

SUPER

L. 7.500 - Fr.s 11,25
Marzo - Aprile 1987 - Anno 4 - N. 3/4

COMMODORE 64⁺ e 128

SUPERCOMMODORE 64
La rivista per gli utenti di
COMMODORE 64 e 128

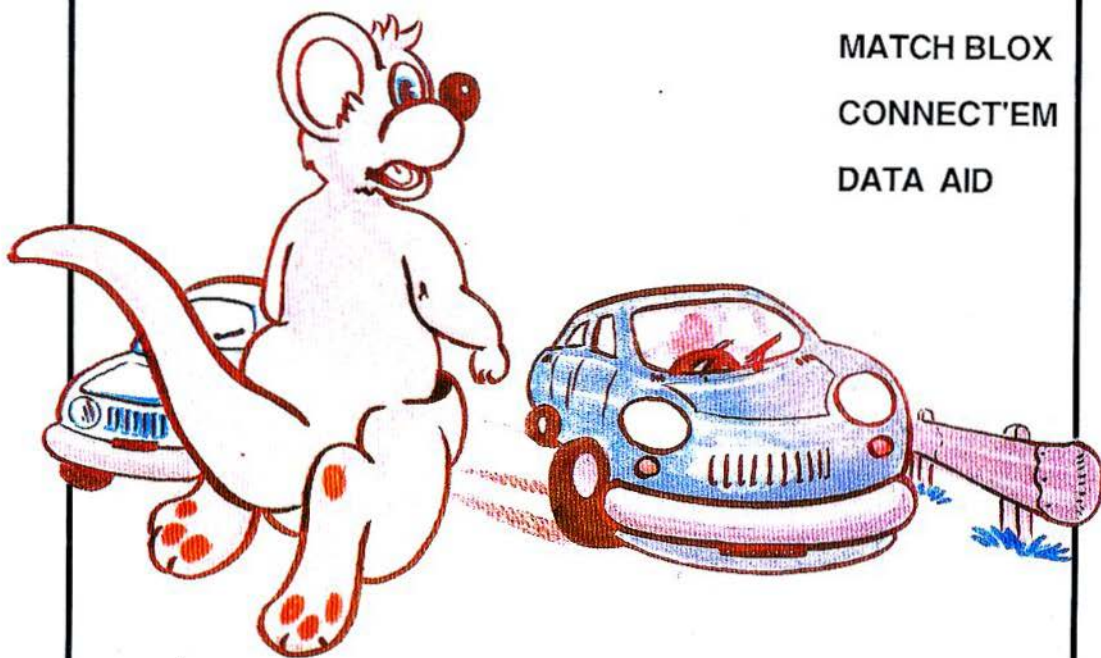
COME MODIFICARE LE ICONE DEL GEOS
MULTITASKER

META BASIC PLUS PER C64

MATCH BLOX

CONNECT'EM

DATA AID



in edicola c'è la tua rivista

Noi 128 & 64 per capire e conoscere il tuo computer. Per lavorare, studiare e giocare. In questo numero troverai il foglio elettronico BUSICALC e un magnifico supergioco per provare la tua abilità "BOMBO". Ma non è finita: potrai scoprire memorie misteriose e imparare l'astronomia, e non è ancora tutto infatti per VIDEOMUSIC suonerai Gloria e tanta tanta grafica facile per i tuoi programmi colorati. In 128 & 64 non solo computer ma anche scuola, giochi, cultura, musica e telecomunicazioni. Il tutto naturalmente firmato Gruppo Editoriale Jackson.

la vuoi magazine, magazine o magazine?



SUPER

COMMODORE 64 e 128

4 **&WRITE**
La posta dei lettori

6 **OPEN**

La creazione di grafici
a istogrammi con il C64
di R. Jacob
trad. e adatt.
di T. Farina

8 **LA CULTURA DELL'INFORMATICA**
...then
Un'introduzione ai compilatori
di T. Heinareck
trad. e adatt.
di M. Cristuib Grizzi

14

14 Match Blox
di M. Tuttle
trad. e adatt.
di S. Colombo

18 Q-Bird
di M. Sedore
trad. e adatt.
di S. Colombo

26 Connect'Em
di F. Chambers
trad. e adatt.
di E. Stefani

29 Decipede
di J. Knesek
trad. e adatt.
di S. Colombo

39 **WORK**

39 Data Aid
di H. Pesch
trad. e adatt.
di S. Colombo

43 Come modificare le
icone del GEOS
di D. Curtis
trad. e adatt.
di R. Polini

46 Multitasker
di R. Daley
trad. e adatt.
di E. Stefani

50 Basic examiner
di R. Kepahart
trad. e adatt.
di S. Colombo

54 MetaBasic Plus per C64
di K. Mykytyn
trad. e adatt.
di M. Cristuib Grizzi

GRUPPO EDITORIALE
JACKSON S.p.A.
San Francisco - Londra -
Madrid - Milano

DIREZIONE, REDAZIONE,
AMMINISTRAZIONE
Via Rosellini, 12
20124 MILANO
Tel. (02) 68.80.951/2/3/4/5
Telex 333436 GEJIT I

SEDE LEGALE
Via G. Pozone, 5 - 20121 Milano

CONSOCCIALE ESTERE USA
GEJ Publishing Group, Inc.
a subsidiary of
Gruppo Editoriale Jackson
1307 S. Mary Avenue - 94087
Sunnyvale CA
Tel. (408) 7730138 - Telex 49959972

GRAN BRETAGNA
GEJ Publishing Ltd.
167 Oxford Street London W1R1AJ
Tel. (01) 4392931 - Telex (051)
21248

DIREZIONE EDITORIALE
Danielle Comboni

DIREZIONE AMMINISTRATIVA
Giuliano Di Chiano

DIREZIONE DIVISIONE LIBRI
E GRANDI OPERE
Roberto Pancaldi

DIRETTORE RESPONSABILE
Giampietro Zanga

REDAZIONE
Mauro Cristuib Grizzi

HANNO COLLABORATO A
QUESTO NUMERO

E. Stefani P. Arcella S. Colombo

P. Dell'Acqua C. Marrelli

GRAFICA E IMPAGINAZIONE
Ivana Rossi

STAMPA
Litografia del Sole
20080 Albairate (MI)

PUBBLICITA'
Concessionaria per l'Italia e l'Estero
J. Advertising s.r.l.
Viale Restelli, 5 - 20124 Milano
Tel. (02) 68.82.895-68.82.48-
68.80.606
TELEX 316213 REINA I

Concessionaria per USA e Canada
International Media Marketing
16704 Marguard Avenue
P.O. Box 1217 - Cerritos CA 90701
Tel. (213) 926-9552

DISTRIBUZIONE
SDOIP
Via Zuretti, 25 - 20125 Milano
Spedizione in abbonamento postale
Gruppo IV/70
Pubblicità inferiore al 70%
Prezzo della rivista L. 7.500
Numero arretrato L. 15.000
Abbonamento L. 82.500 (11 numeri
con cassetta); L. 38.500 (11 numeri
senza cassetta)
per l'estero L. 110.000 (11 numeri
con cassetta); L. 50.000 (11 numeri
senza cassetta)

I versamenti vanno indirizzati a:
Gruppo Editoriale Jackson
Via Rosellini, 12 - 20124 Milano
mediante emissione di assegno
bancario, vaglia o utilizzando il
C.C.P. n° 11668203
Per i cambi di indirizzo indicare, oltre
al nuovo, anche l'indirizzo
precedente e allegare L. 500, anche
in francobollo.

AUTORIZZAZIONE ALLA
PUBBLICAZIONE
Tribunale di Milano
n° 155 del 5/4/86

GLI ARTICOLI TRADOTTI SONO
TRATTI DALLE RIVISTE
COMPUTER DA
COMPUTER GAZETTE
COMPUTER PUBLICATION, INC.
P.O. Box 5406, Greensboro, NC
27403 - USA

Il Gruppo Editoriale Jackson è
iscritto nel Registro nazionale
della Stampa al n° 117 vol. 2 -
foglio 129 in data 17-8-1982



Rivista associata
all'Unione
Stampa
Periodica
Italiana



Associata

READ & WRITE



Riposizionare i file sulla directory

*Come posso fare in modo che l'ordine con cui appaiono i programmi sulla directory del disco venga modificato? Ciò mi serve per fare in modo che i file di utilizzo più comune appaiano per primi e possano così essere caricati tramite LOAD***,8.*

F. Trimino

R. Non è esatto che l'istruzione LOAD***,8 provochi il caricamento del primo programma listato nella directory: in realtà questa istruzione carica il file di più recente accesso. Ciò significa che se si salva un file su disco e poi si digita LOAD***,8, verrà ricaricato proprio quel file.

Tra l'altro, questa caratteristica risulta particolarmente utile se si è per errore cancellato un file da disco: sarà sufficiente, se non si sono fatte altre operazioni su disco dopo lo "scratch", digitare LOAD***,8 e risalire normalmente il file.

Per caricare il primo file nell'ordine del listato della directory del disco, occorre invece utilizzare il comando LOAD***,1.

E' possibile cambiare l'ordine di apparizione dei file utilizzando appositi programmi che permettono anche di ordinare alfabeticamente i programmi su disco. La modifica manuale dell'ordine dei file, pur essendo possibile, è alquanto macchinosa e non ci sentiamo di consigliarla.

Indirizzi con SpeedScript

Posseggo un C64 con stampante Commodore 1526, e utilizzo con soddisfazione SpeedScript con la carta a modulo

continuo. Ho però dei problemi con la carta a fogli singoli e nella stampa degli indirizzi sulle buste: potete suggerirmi il modo corretto per eseguire queste operazioni?

L. Angelini

R. Utilizzando SpeedScript con carta a fogli singoli occorre inserire - all'inizio del documento - il comando di attesa per ogni salto pagina. Tale comando si ottiene premendo contemporaneamente i tasti CTRL e £ (lira) seguiti dal tasto W. In questo modo sarà possibile inserire il nuovo foglio per ogni pagina di stampa. La stampante 1526 è dotata di un sensore di fine carta che, utilizzando i fogli singoli, provoca l'arresto della stampa poiché la stampante "pensa" che la carta sia esaurita; con il comando sopracitato il sensore non entra in funzione e la stampa a foglio singolo può essere effettuata agevolmente.

La stampante non sarà tuttavia in grado di stampare gli indirizzi sulle buste proprio a causa del sensore di fine carta: la busta non è sufficientemente lunga per poter "ingannare" il sensore.

L'unico modo che ci viene in mente per aggirare questo problema consiste nell'attaccare la busta a un normale foglio di carta mediante un po' di nastro adesivo: il sensore potrà essere ingannato e, posizionando adeguatamente la busta nel rullo, si potrà stampare l'indirizzo.

Certamente non è questa una procedura né semplice né rapida, e va bene soltanto nel caso si abbia un solo indirizzo da stampare.

Più comodo è l'acquisto di carta a modulo continuo già dotata di etichette autoadesive per indirizzi che vengono quindi agevolmente applicate sulle buste. Senza effettuare la spesa per tale tipo di carta è anche possibile stampare nor-

malmente l'indirizzo, ritagliandolo quindi dalla carta e incollandolo sulla busta.

Collegamento home-personal

Possedendo in ufficio un Personal Computer IBM e, per la casa, un C64, mi chiedevo se fosse possibile utilizzare a casa il C64 per la stesura di relazioni e lettere, e poi trasferire i testi sull'IBM per archivarli nel suo disco rigido e stamparli con la stampante di qualità che vi è collegata. Potete aiutarmi?

R. Agostini

R. Il C64 e l'IBM PC o XT (e in generale tutti i computer MS-DOS) non sono in alcun modo compatibili: un dischetto formattato dal C64 non è leggibile dal MS-DOS, e viceversa.

L'unico modo per trasferire dei testi da un sistema all'altro è il collegamento diretto via porta seriale RS-232 (che è standard sul MS-DOS e richiede un adattatore sul C64) oppure via modem e telefono.

E' richiesto che i testi scritti con il C64 siano inviati alla RS-232 in forma ASCII sequenziale, pertanto si deve utilizzare un programma di word processing che permetta di salvare i testi in forma sequenziale (per esempio Easy Script) oppure occorre un programma per convertire i testi in forma sequenziale. Il nostro SpeedScript è dotato di un convertitore (separato) che permette di convertire i testi in vero ASCII e quindi di inviarli alla RS-232.

All'altro capo del filo, il computer MS-DOS dovrà essere dotato di un programma di comunicazione (per esempio il diffuso Lotus Symphony) in grado di ricevere i testi e salvarli sul disco rigido.

OK-LIST

Sappiamo tutti quanto sia difficile digitare correttamente un programma (ovviamente, se non si dispone della cassetta allegata alla rivista). Errori di digitazione anche banali possono provocare il malfunzionamento del programma stesso o anche il blocco completo del computer, obbligando a riprendere da capo tutto il lavoro.

OK-LIST è un particolare programma in grado di intercettare gli errori commessi durante la digitazione di uno dei listati da noi pubblicati, e di segnalarveli.

Digitate il listato pubblicato in questa pagina (attenzione a non commettere errori: OK-LIST non può controllare se stesso!) e quindi salvatelo su nastro o disco.

Quando dovrete accingervi a digitare uno dei nostri programmi in BASIC, caricate per prima cosa OK-LIST e dategli il RUN: dopo una breve attesa, necessaria a inserire in memoria una routine in linguaggio macchina, comparirà sullo schermo il messaggio "OK-LIST ATTIVATO".

Potete a questo punto iniziare a ricopiare il programma BASIC prescelto, evitando di digitare la "rem" seguita da un numero che compare al termine di ogni linea. Come vedrete, alla pressione di RETURN apparirà un numero nell'angolo superiore sinistro dello schermo; confrontatelo con quello riportato dopo la "rem" sul listato: se è diverso, avete commesso un errore nel digitare quella linea di programma.

OK-LIST non verifica gli spazi all'interno di una linea di istruzione: se tutti i checksum (così vengono chiamati i numerini di controllo visualizzati da OK-LIST) sono corretti, e il programma genera invece delle videate "spezzettate", probabilmente avete dimenticato o aggiunto qualche spazio di troppo all'interno delle virgolette di una istruzione PRINT.

Inoltre, OK-LIST non è in grado di controllare la sequenza con la quale vengono digitati i caratteri all'interno di un'istruzione; ciò è particolarmente pericoloso nel caso delle istruzioni DATA contenenti molti numeri. Ad esempio, digitando 220 dove andrebbe digitato 202, non si verifica alcuna variazione nel checksum, pur avendo commesso un errore; consi-

gliamo quindi di porre grande attenzione nel digitare tale tipo di istruzioni.

Un'ultima avvertenza sull'utilizzo di OK-LIST riguarda le abbreviazioni delle istruzioni BASIC: esse non vengono riconosciute da OK-LIST e generano pertanto un checksum differente. Se si utilizza OK-LIST, le istruzioni BASIC dovranno quindi essere digitate per intero.

Data la sua posizione in memoria (il buffer del registratore), OK-LIST interferisce con ogni operazione con l'unità a cassette. Per questo motivo prima di effettuare un SAVE su nastro OK-LIST dovrà essere preventivamente disabilitato attraverso la pressione di RUN/STOP e RESTORE.

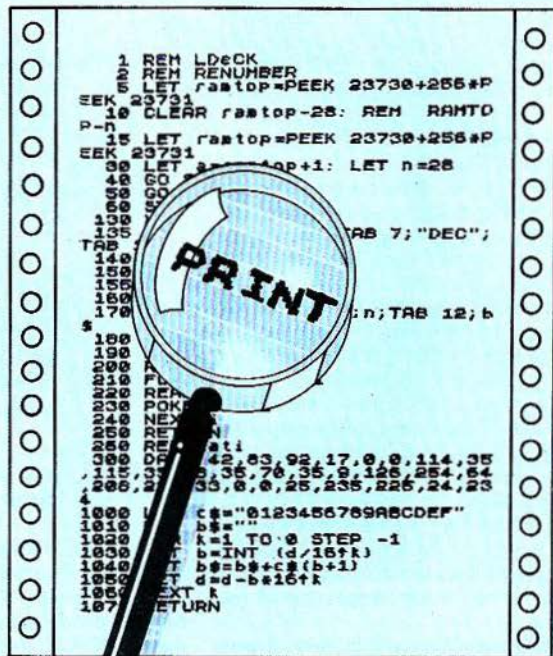
Non è possibile effettuare operazioni di LOAD da cassetta senza cancellare OK-LIST dalla memoria.

Queste limitazioni non sono invece presenti, se si utilizza il disk drive come memoria di massa.

Nel caso si preme accidentalmente RUN/STOP e RESTORE durante l'utilizzo di OK-LIST, esso potrà essere riattivato digitando SYS886.

Buon lavoro!

```
10 PRINT "{CLR}ATTENDERE PREGO..."
11 FOR I=886 TO 1018: READ A:CK=CK+A:POKE I,A:NEXT
12 IF CK<>17539 THEN PRINT "{GIU'}ERRORE NELLE ISTRUZIONI DATA":END
13 SYS886:PRINT "{CLR}{ 2 GIU'}OK-LIST ATTIVATO.":NEW
14 DATA 173,036,003,201,150,208
15 DATA 001,096,141,151,003,173
16 DATA 037,003,141,152,003,169
17 DATA 150,141,036,003,169,003
18 DATA 141,037,003,169,000,133
19 DATA 254,096,032,087,241,133
20 DATA 251,134,252,132,253,008
21 DATA 201,013,240,017,201,032
22 DATA 240,005,024,101,254,133
23 DATA 254,165,251,166,252,164
24 DATA 253,040,096,169,013,032
25 DATA 210,255,165,214,141,251
26 DATA 003,206,251,003,169,000
27 DATA 133,216,169,019,032,210
28 DATA 255,169,018,032,210,255
29 DATA 169,058,032,210,255,166
30 DATA 254,169,000,133,254,172
31 DATA 151,003,192,087,208,006
32 DATA 032,205,189,076,235,003
33 DATA 032,205,221,169,032,032
34 DATA 210,255,032,210,255,173
35 DATA 251,003,133,214,076,173
36 DATA 003
```





OPEN



di R. Jacob
trad. e adatt. di T. Farina

La creazione di grafici a istogramma con il C64

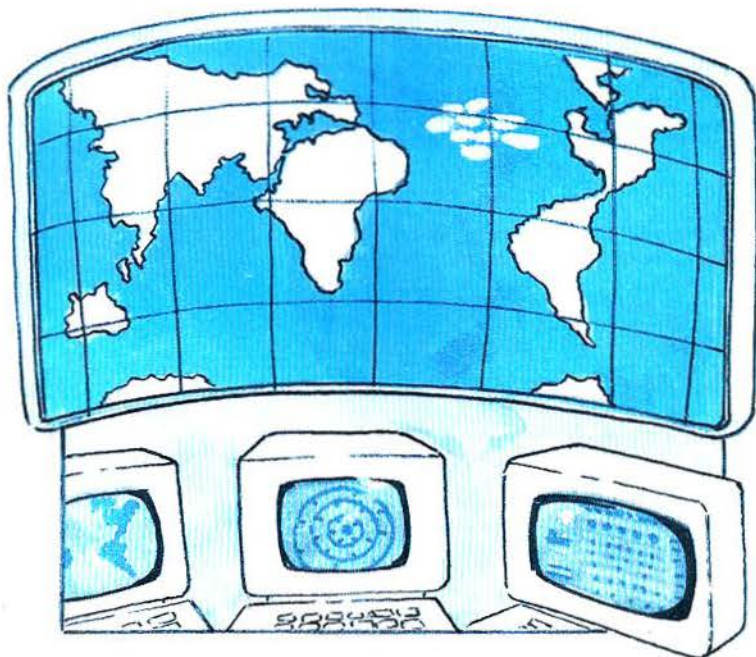
Il C64, nato come home computer dedicato soprattutto ai videogiochi, si è poco per volta inserito anche nel settore delle piccole applicazioni commerciali e in qualche ufficio. Per questo motivo, vediamo in questo articolo come è possibile realizzare con il C64 la cosiddetta "business graphics", ovvero la rappresentazione di dati numerici sotto forma di grafico.

Chi utilizza il C64 anche per piccole applicazioni produttive, come per esempio la gestione del conto corrente bancario o la memorizzazione dei dati relativi alle vendite, si sarà certamente reso conto che l'interpretazione dei dati viene enormemente semplificata riducendo questi ultimi in forma grafica piuttosto che come tabulato numerico.

Un grafico "a salire" ci fa immediatamente vedere che il nostro conto in banca gode di ottima salute, oppure che le vendite stanno migliorando: il messaggio è molto più intuitivo di quello criptico dato da una sequenza di numeri stampati dal computer.

In questo articolo vedremo come è possibile realizzare con il C64 un grafico a istogramma, partendo dai dati numerici elaborati dal nostro programma di gestione.

Il programma proposto in fondo a questo articolo e chiamato "Bar Charter" può infatti essere aggiunto, come su-



broutine, al nostro programma principale dotandolo così di un'interessante capacità grafica.

Desiderando utilizzare Bar Charter in questo modo, cancellate le linee con numero inferiore a 1000 ed effettuate un MERGE con il vostro programma. Il comando MERGE non è presente nel BASIC del C64, tuttavia molte utility di programmazione permettono di aggiungere

questo comando (per esempio METABASIC da noi precedentemente pubblicato) al BASIC standard.

Il programma principale dovrà immagazzinare i valori numerici da rappresentare in forma grafica (al massimo 38 valori) in un vettore di nome B(), che verrà quindi dimensionato con una DIM B(38).

La visualizzazione del grafico verrà attivata in questo caso con un semplice

GOSUB 1000.

Se si desidera invece inserire i dati numerici da rappresentare in forma grafica direttamente da tastiera, il programma può essere utilizzato direttamente così com'è: i valori da rappresentare vanno inseriti nelle linee DATA da 120 a 160 al posto di quelli presenti per scopi dimostrativi.

Dando il RUN al programma senza modificare i dati numerici si può ottenere una

dimostrazione delle capacità della routine: vengono visualizzati 38 valori sotto forma di istogrammi; la scala delle ordinate viene calcolata in base al coefficiente di scala. Tale coefficiente di scala viene richiesto dalla routine, e deve essere tale che la sua divisione per il valore più elevato da visualizzare dia come risultato un numero minore di 20. Se questa regola non viene rispettata, alcune barre dell'istogramma verranno sostituite dal simbolo ">" per indicare che non possono essere visualizzate.

Dopo un minimo di esperienza, sarà facile calcolare il coefficiente di scala più appropriato per l'intervallo di valori da rappresentare graficamente, oppure scrivere per esercizio una breve routine che lo calcoli automaticamente in base al valore più alto e a quello più basso dei nostri dati.

Bar Charter

```

100 N=38:DIMB(N):FORI=1TON:READT:B=B+T:B(I)=B:NEXT      :rem 119
110 INPUT"{CLR}{WHT}NOME GRAFICO";AN$:AN$=LEFT$(AN$,39):PRINT      :rem 242
120 DATA50,20,-15,25,30,12,-20,17      :rem 95
130 DATA5,10,13,-9,22,11,10,-17      :rem 254
140 DATA10,15,12,-30,-25,-10,30,10      :rem 131
150 DATA-10,-25,-16,-27,10,15,-16,10      :rem 236
160 DATA-15,-16,-12,-20,-10,12      :rem 196
1000 REM INIZIO DELLA ROUTINE GRAFICA      :rem 236
1010 POKE 783,0:POKE53280,0:POKE53281,0:P      :rem 32
1020 L5$="5[<4>]"      :rem 80
1030 L0$="0[<4>]"      :rem 71
1040 INPUT"COEFF.DI SCALA";INC:PRINT"      :rem 206
{CLR}"
1050 INC$=STR$(INC):IFLEN(INC$)>5THEN1340      :rem 170
      :rem 148
1060 IFINC<1THEN1340
1070 PRINT"COEFFICIENTE DI SCALA      :rem 5
{ 9 SPAZI}="INC
1080 FORI=1TO39:PRINT"RVS ";:NEXT:PRINT      :rem 69
:PRINT"{SU}{RVS}"AN$"{OFF}"
1090 FORI=1TO4:PRINT"OFF":NEXT:rem 102
1100 PRINTL5$:FORI=1TO4:PRINT"OFF":NEXT      :rem 202
      :rem 198
1110 PRINTL0$:FORI=1TO4:PRINT"OFF":NEXT
1120 PRINTL5$:FORI=1TO4:PRINT"OFF":NEXT      :rem 204
      :rem 240
1130 VL=22:X=1:A=0      :rem 114
1140 A=A+1:IFA>NTHEN1290      :rem 226
1150 COL=COL+1:IFCOL>15THENCOL=1:rem 161
1160 POKE646,COL      :rem 113
1170 BL=B(A)/INC      :rem 176
1180 IFINC>B(A) THENEM=1:GOTO1260:rem 49
1190 IFBL=>20 THENEM=2:GOTO1260      :rem 51
1200 DV=VL-INT(BL)      :rem 34
1210 POKE782,X:POKE781,DV:SYS65520      :rem 55
      :rem 72
1220 PRINT"OFF"      :rem 153
1230 IFBL<2THEN1250
1240 FORI=1TOBL-1:PRINTTAB(X)"OFF":NEXT      :rem 74
      :rem 240
1250 X=X+1:GOTO1140
1260 POKE782,X:POKE781,21:SYS65520      :rem 208
      :rem 21
1270 IFEM=1THENPRINT"<":GOTO1250:rem 204
1280 PRINT">":GOTO1250
1290 POKE782,0:POKE781,22:SYS65520
1300 PRINT"<4>{RVS}{ 2 SPAZI}<-SOTTO SC      :rem 196
ALA{ 3 SPAZI}:{ 3 SPAZI}>-SOPRA SCAL
A{ 2 SPAZI}{OFF}"
1310 PRINT"UN TASTO PER USCIRE"      :rem 118
1320 GETOP$:IFOP$=""THEN1320      :rem 107
1330 END:REM [FINE ROUTINE - RITORNO ALPR      :rem 48
OGRAMMA PRINCIPALE]
1340 PRINT"{GIU'}VALORE ERRATO - FUORI SC      :rem 159
ALA{GIU'}":GOTO1040

```





Un'introduzione ai compilatori

di T. Heimarck

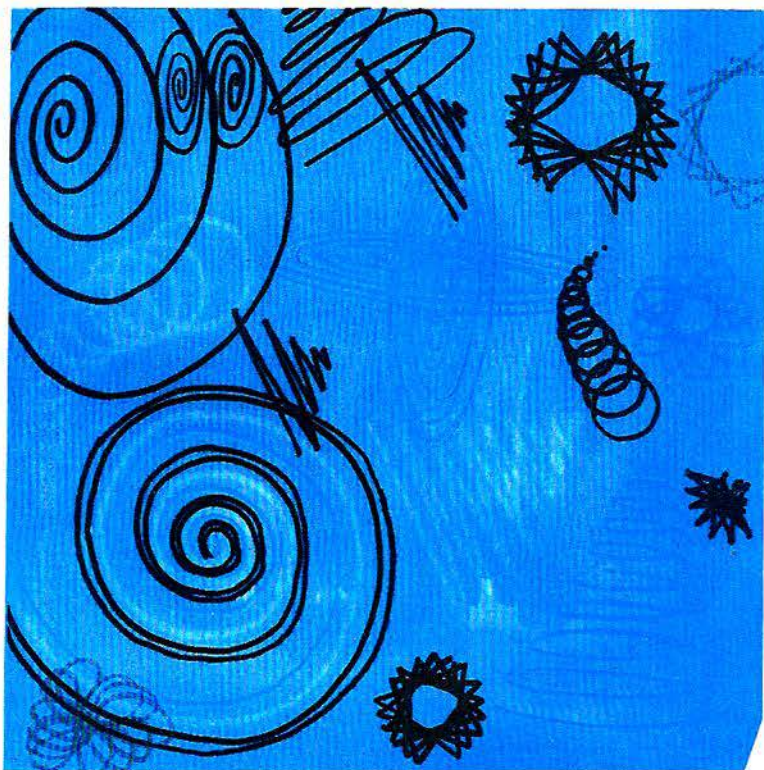
trad. e adatt. di M. Cristuib Grizzi

Un compilatore consiste in una potente utility in grado di tradurre programmi scritti in un linguaggio ad alto livello - come il BASIC - in linguaggio macchina immediatamente comprensibile dal computer senza la necessità di essere interpretato. In questo articolo esamineremo il modo di funzionamento dei compilatori BASIC.

Il linguaggio di programmazione BASIC ha molti pregi, ma anche un principale difetto: la lentezza nell'esecuzione dei programmi. Supponiamo che abbiate lavorato a lungo e di buona lena per scrivere un lungo e complesso programma BASIC: potrebbe essere un gioco "arcade" oppure un programma database per manipolare fino a un migliaio di record per applicazioni domestiche. Il programma funziona perfettamente, e non vengono segnalati errori durante la sua esecuzione; esso è tuttavia lento: gli SPRI-TE non si muovono sullo schermo con la rapidità desiderata, oppure la ricerca di un dato record tra mille può richiedere minuti di attesa... In questo caso, avete raggiunto il "limite di velocità" del BASIC del vostro computer.

Infatti il BASIC non offre le migliori prestazioni in termini di velocità, soprattutto se si tratta di muovere immagini sullo schermo oppure di eseguire degli ordinamenti ed estrazioni in una grande quantità di stringhe diverse.

Supponiamo ora che, in cerca di consigli per ottenere una maggiore velocità di esecuzione, andiate con il vostro programma da tre amici dotati di un'esper-



ienza di programmazione superiore alla vostra. Il primo di essi, esperto soprattutto della programmazione in BASIC, vi consiglierà per esempio: "Credo che questa subroutine debba essere riscritta in modo più efficiente; questi vettori devono essere eliminati e trasformati in una matrice tridimensionale. Inoltre, l'algoritmo di ordinamento usato non è dei più veloci e va modificato in un quick sort... OK, dammi una settimana o due e ti modificherò il programma in modo che 'giri' più velocemente almeno del

venti per cento!"

Il secondo esperto, abituato soprattutto a programmare direttamente in linguaggio macchina, vi dirà: "Non ci sono trucchi e modifiche che tengano: se il programma è in BASIC sarà sempre lento. Posso, se vuoi, riscriverlo in linguaggio macchina, e otterrai senz'altro un'esecuzione a razzo... Naturalmente mi occorre almeno un mese di tempo per effettuare la conversione!"

Il vostro terzo amico, che 'smanetta' da tempo con computer e programmi, vi di-

ce invece: "Ho capito qual'è il problema, aspetta un attimo..." Dopo pochi minuti ritorna con un dischetto in mano, lancia un programma, risponde a qualche INPUT, sostituisce il dischetto nel drive inserendo il vostro e, dopo circa due minuti, vi restituisce una versione del vostro programma almeno quattro volte più veloce. Come ha fatto? Il vostro programma è stato compilato.

Dove nasce la velocità

Un compilatore BASIC può aumentare la velocità di esecuzione di un programma riscrivendolo in una forma consistente praticamente in puro linguaggio macchina.

Il termine "compilatore BASIC" non indica che il programma compilatore sia scritto in BASIC (non lo è anzi praticamente mai), bensì che si tratta di un compilatore destinato a operare su programmi BASIC. Analogamente ai compilatori BASIC, esistono compilatori per i linguaggi Pascal, FORTRAN, C, eccetera.

Consideriamo il comando PRINT: la sua funzione è molto semplice, ovvero visualizzare qualcosa sullo schermo. In verità, il comando PRINT può effettuare una vasta serie di operazioni:

```
PRINT
PRINT "CIAO"
PRINT A
PRINT ABS(A+B)
PRINT "IL NUMERO"; LOG(VAL(A)); " E'
IL LOGARITMO NATURALE DI "; A$
```

Quando, durante l'esecuzione di un programma, viene incontrata un'istruzione PRINT, l'interprete BASIC deve analizzarne il contesto. E' una PRINT singola non seguita da alcun argomento? E' seguita da una stringa racchiusa fra virgolette? Si riferisce a una variabile? A una funzione? Oppure a una serie di stringhe, variabili e funzioni separate da punto e virgola?

E' possibile pensare a PRINT come a un comando singolo, in realtà esso ha molti aspetti tra loro diversi: prima che il computer esegua un'istruzione PRINT, deve andare a leggere il resto della linea di programma per controllare quale tipo di PRINT deve essere effettuato. Se ci sono delle parentesi, il procedimento diventa ancora più complicato: il computer deve ricostruire il significato della linea iniziando dalla parentesi aperta più interna, per finire con l'ultima parentesi chiusa.

Esiste inoltre un secondo livello di controllo, nel caso siano presenti delle variabili. Dopo che il computer abbia stabilito che PRINT AB debba dare come effetto la visualizzazione del contenuto della variabile chiamata AB, occorre andare a cercare in memoria questa variabile, e

leggerne il contenuto.

Se sono state definite altre 57 variabili prima che AB venga utilizzata per la prima volta, essa si troverà in cinquantottesima posizione nelle celle di memoria del computer. L'istruzione PRINT dovrà passare quindi in rassegna 57 variabili che non interessano prima di localizzare il contenuto di AB.

Un lavoro da interpreti

Il procedimento di analisi di una linea durante l'esecuzione di un programma (stabilendo l'operazione che deve essere effettuata da ogni singolo comando, analizzandone il contesto, ricercando eventuali variabili, e così via) è chiamato *interpretazione*.

Durante l'esecuzione di un programma BASIC, ogni comando eseguito deve prima essere interpretato: per questo esiste l'interprete BASIC, e per questo il programma non può avere una velocità elevata.

Un compilatore permette di accrescere la velocità di esecuzione interpretando i comandi del programma prima che questo venga eseguito.

Per esempio, un compilatore può esaminare un comando come PRINT AB e assegnare una piccola porzione di memoria alla variabile AB; supponiamo che la variabile AB vada a occupare la locazione 5003. Il compilatore immagazzinerà quindi nel programma compilato (o *programma oggetto*) un comando che sarà simile a:

```
PRINT (il valore nella locazione 5003)
```

Invece di dover cercare durante l'esecuzione del programma la locazione di memoria in cui si trova il contenuto della variabile, il compilatore "si segna" questo indirizzo una volta per tutte durante il processo di compilazione. Quando il programma oggetto verrà eseguito, il computer potrà leggere direttamente il contenuto di quella locazione, senza dover effettuare ricerche per la memoria.

Come ulteriore esempio, consideriamo la linea $A=C*34.85$, che moltiplica la variabile C per la costante 34.85, e assegna il risultato alla variabile A.

In un programma BASIC, il computer dovrà andare a cercare il numero contenuto in C, e quindi andare a ritrovare l'equivalente in virgola mobile dei cinque caratteri "34.85" (il formato a virgola mobile è un modo particolare per rappresentare i numeri, particolarmente utile per l'effettuazione di operazioni matematiche).

I due numeri dovranno quindi essere tra loro moltiplicati inserendo il risultato dell'operazione nella variabile A.

Un compilatore deve eseguire le stesse operazioni, anche se la maggior parte di esse vengono eseguite in fase di compila-

zione e non in fase di esecuzione del programma.

Il compilatore ricava il valore in virgola mobile di 34.85, e stabilisce quali locazioni assegnare alle variabili A e C.

In fase di esecuzione del programma oggetto, il tempo richiesto per l'operazione matematica sarà di molto inferiore in quanto il computer non dovrà perdere tempo traducendo formati numerici e ricercando variabili.

Il compilatore ha già fatto la maggior parte del lavoro: l'interpretazione.

Ancora un esempio: un comando come GOTO 780 produce l'attivazione di parecchie procedure; per prima cosa, il computer traduce i tre caratteri "780" in un binario intero (nel formato byte basso/byte alto), dal momento che i numeri di linea vengono immagazzinati in memoria con questo formato. Questo numero viene confrontato con il numero della linea eseguita in quel momento: se quest'ultimo è superiore, il computer ritorna alla prima linea del programma e inizia la ricerca della linea 780. Se, al contrario, il numero della linea eseguita in quel momento è inferiore a 780, il computer inizia la ricerca a partire dal punto della memoria in cui si trova. Solo al termine di queste operazioni il controllo viene passato alle istruzioni contenute nella linea 780.

Vediamo ora come un GOTO viene trattato da un compilatore: anch'esso, naturalmente, deve tradurre il numero 780 nel formato binario intero, e localizzare la posizione in memoria di questa linea. Dopo averlo fatto, esso inserisce un'istruzione di salto in linguaggio macchina (JMP) in modo che il controllo passi direttamente alla serie di istruzioni che corrispondono alla linea 780.

Anche qui, tutto il lavoro viene fatto in fase di compilazione, mentre in fase di esecuzione si avrà un semplice JMP alla locazione specificata.

Qualche limitazione

Naturalmente anche i compilatori, pur comportando grandi vantaggi in termini di velocità di esecuzione dei programmi, presentano qualche inconveniente.

Il primo dei problemi è che alcuni compilatori sono compatibili soltanto con un sottoinsieme dei comandi BASIC. Esistono alcune istruzioni che non vengono riconosciute dal compilatore, e quindi esse non possono venire utilizzate all'interno del programma. Con l'evoluzione del software per C64, quasi tutti i compilatori commerciali sono tuttavia ormai compatibili con tutti i comandi del suo BASIC.

Alcuni compilatori, inoltre, producono del codice oggetto estremamente veloce quando si effettuano operazioni di matematica intera (cioè soltanto con numeri interi, senza decimali), mentre la

velocità decade rapidamente se si incontrano operazioni con decimali. Anche questo difetto dipende dal tipo di compilatore utilizzato.

La migliore tecnica di utilizzo dei compilatori consiste nello scrivere i programmi in un linguaggio di alto livello come il BASIC, potendo effettuare correzioni e modifiche in qualsiasi momento e, soprattutto, provando il funzionamento delle singole subroutine del programma. La programmazione in BASIC è molto semplice, e non richiede particolare esperienza sulla struttura dell'hardware del proprio computer.

Una volta terminata la stesura del programma in BASIC (detto *programma sorgente*) e verificato a fondo il suo corretto funzionamento, esso potrà essere compilato. Il compilatore legge il programma sorgente, lo interpreta, e contemporaneamente scrive su disco il programma oggetto in linguaggio macchina. Una volta compilato, il programma sorgente potrà essere archiviato, e si potrà caricare e mandare in esecuzione il programma oggetto. Quest'ultimo non potrà chiaramente più essere modificato: se si volessero, per esempio, variare delle caratteristiche anche semplici, occorrerà agire sul programma sorgente e ripetere la compilazione in modo da ricavare un nuovo programma oggetto con le caratteristiche desiderate.

Alcuni linguaggi sono disponibili soltanto in versione compilata: ciò significa che non si potrà provare il programma sorgente prima di eseguire la compilazione. In genere, in questi casi il programma sorgente viene scritto con qualunque word processor in grado di creare file ASCII sequenziali; sarà soltanto in fase di compilazione che ci verranno segnalati eventuali errori di sintassi presenti nel programma.

Un'altra caratteristica negativa dei compilatori consiste nel fatto che il programma oggetto è quasi sempre molto più lungo del programma sorgente: il programma oggetto deve spesso comprendere infatti anche una consistente collezione di routine in linguaggio macchina (detta *libreria*) necessaria in fase di esecuzione, e aggiunta automaticamente in fase di compilazione.

Naturalmente, un programma compilato non avrà mai l'elevata velocità che si otterrebbe traducendo manualmente in linguaggio macchina il programma BASIC: ciò perché il procedimento di compilazione segue schemi fissi che si devono adattare alla maggior parte delle evenienze possibili, e contengono quindi una notevole ridondanza di istruzioni macchina. Un programmatore esperto di linguaggio macchina sarebbe certamente molto più abile di qualunque compilatore, sapendo esattamente quale sia il risultato che ci si aspetta di ottenere dal programma!

L'accento che abbiamo fatto prima alle librerie comporta un'ulteriore distinzione: esistono compilatori che creano programmi oggetto totalmente in linguaggio macchina e non richiedono librerie, e ne esistono altri che creano programmi scritti nel cosiddetto *pseudocodice* (un linguaggio simile al linguaggio macchina ma con minore efficienza) e richiedono quindi l'uso delle librerie (dette più comunemente *RUN-TIME*) per la loro esecuzione.

Nella tabella si trova un elenco dei principali compilatori per C64 e C128 esistenti: molti tra essi si trovano facilmente anche sul mercato italiano.

ADA Compiler

compilatore della Abacus Software per il linguaggio ADA (USA)

BASIC Compiler

compilatore BASIC della Abacus Software in versioni separate per C64 e C128 (USA)

Blitz!

compilatore BASIC della Skyles Electric Works (USA)

C Power

compilatore per linguaggio C della Pro-Line Software (USA)

COBOL Compiler

compilatore per linguaggio COBOL della Abacus Software (USA)

C-64 Forth/79

compilatore per linguaggio Forth della Performance Micro Products (USA)

Forth-64 Language

compilatore per linguaggio Forth della Abacus Software (USA)

Gnome Speed

compilatore BASIC per C128 della SM Software (USA)

KMM Pascal

compilatore Pascal della Wilserve Industries (USA)

Kyan Pascal

compilatore Pascal della Kyan Software (USA)

MasterForth

compilatore Forth con sistema grafico della MicroMotion (USA)

PROMAL

compilatore per linguaggio PROMAL della SMA (USA)

The Sixty Forth

compilatore Forth della Elcom Publishing (USA)

Super C Language Compiler

compilatore C in due versioni per C64 e C128 della Abacus Software (USA)

Superforth 64

compilatore Forth della Parsec Research (USA)

Superforth 64 + Artificial Intelligence

compilatore Forth con pacchetto di sviluppo per sistemi esperti della Parsec Research (USA)

Super Pascal

compilatore Pascal in versioni separate per C64 e C128 della Abacus Software (USA)

White Lightning

compilatore Forth della Oasis Software (USA)



CERCHIAMO COMPUTER STANCHI DEL SOLITO MENU.



NUOVO SAPORE D'AVVENTURA

C64 e C128 è ora di cambiare!

Ecco a voi i Jacksoniani: Jack, Sonny e Ann tre intrepidi eroi della galassia di Lug che difendono L'Armonia Cosmica dalle Forze Oscure. È facilissimo entrare nel loro mondo basta un "load" ed eccoti protagonista di una meravigliosa avventura galattica e con un semplice "fine" salti con tranquillità da un'avventura ad un'altra, da un pianeta ad un asteroide. ...Gli abitanti della terra trasferiti nella lontana galassia di Lug lottano per difendere l'Armonia Cosmica tutelata dal Patto Delta. Ma le Forze Oscure tentano di turbare questa serenità approfittando anche della Lebbra Cosmica che contagia indifferentemente persone e cose "gettando nel panico" gli abitanti; niente paura però perché ci sono Jack, Sonny e Ann. Jack impavido e valoroso guerriero ha una missione di grandissima importanza da compiere... Sonny, fuggito da Larus sull'asteroide Xart cerca di salvare gli abitanti dalla ferocia del tiranno Nubqz, che con il Raggio della Morte vuole annientare i ribelli.

La sua bella clonofiglia Ann viene rapita dalla Nave Nera di Nabqz e la situazione sembra senza speranza. Per fortuna c'è Jack che tenterà di... il resto scopriilo da te con il tuo **C64** o **C128**. Gli ingredienti ci sono tutti: avventura-coraggio-amicizia-perfidia-temerarietà-valore-sorpresa-stupore e, pensa che il protagonista sei soltanto TU!!!

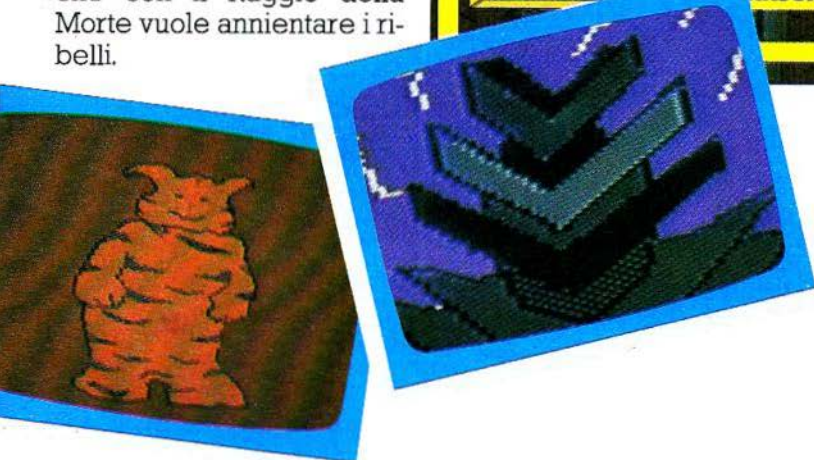
Allora che aspetti? Dai l'input e immergiti nella Galassia, ti aspettano i tuoi amici Jacksoniani. Dal mese di Marzo in edicola un appuntamento mozzafiato: fascicolo

+ cassetta con tre fantastiche adventures che puoi vivere in tre episodi, separatamente, o come parte della stessa storia. A te la scelta! Ma non è tutto: troverai anche i trucchi per diventare campione dei Videogiochi da bar. Nel primo numero i segreti di Dragon's Lair. E le soluzioni alla tua avventura? Ogni numero riporterà le soluzioni dell'avventura precedente per scoprire quanto sei stato abile. Corri in edicola, perché è ora di cambiare.

Buon Adventures!!!



**GRUPPO EDITORIALE
JACKSON**



RA, TUTTI I MESI IN EDICOLA.

Match Blox

di M. Tuttle
trad. e adatt. di S. Colombo

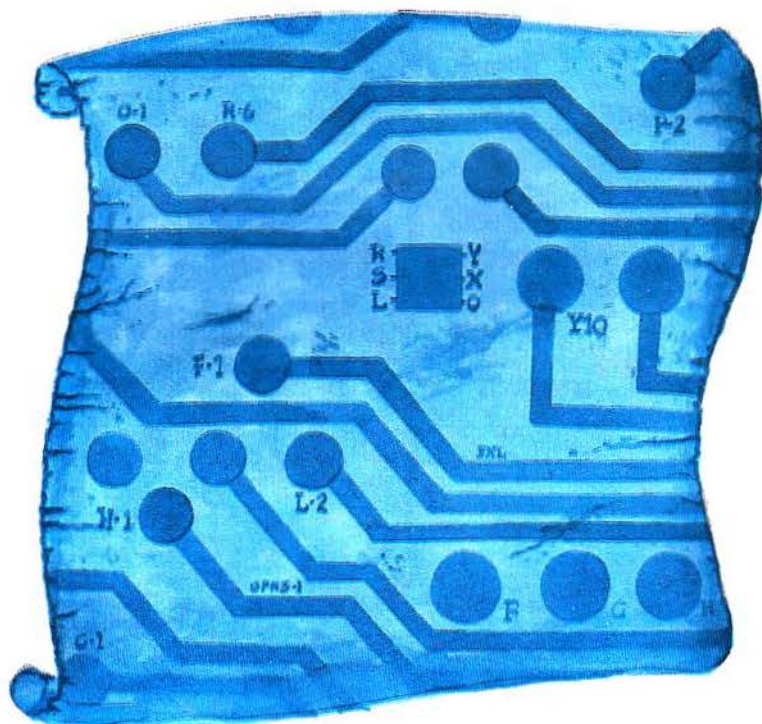
Se vi piacciono i puzzle, questo è il gioco che fa per voi: dotato di cinque variazioni possibili, vi farà passare dei momenti piacevoli alla tastiera del vostro computer.

In Match Blox dovete cercare di risolvere il puzzle presentato dal computer nel minor numero possibile di tentativi.

Ogni partita viene disputata su una griglia formata da tre rettangoli orizzontali e tre verticali, dove ciascun rettangolo è colorato in blu oppure in arancione. A fianco della griglia di gioco viene visualizzata una griglia di dimensioni ridotte, contenente l'obiettivo del puzzle: dovrete infatti, nel minor numero di mosse possibile, muovere i rettangoli della griglia principale fino a formare il disegno mostrato nella griglia più piccola.

Inserite un joystick nella Porta 2 e date il RUN: per prima cosa il programma visualizza i titoli delle cinque variazioni di gioco possibili, numerate da 1 a 5. Un rettangolo nella parte inferiore dello schermo contiene il numero della variazione selezionata, che inizialmente è la numero uno. Se desiderate utilizzare la prima variazione, premete il pulsante di fuoco; se invece desiderate utilizzarne un'altra, muovete il joystick verso destra o verso sinistra fino a che venga visualizzato nel rettangolo il numero relativo alla variazione desiderata, e premete il pulsante di fuoco.

Se immaginiamo che la lettera B rappresenti i rettangoli di colore blu, e la lette-



ra A quelli di colore arancione, le cinque variazioni di gioco possibili sono rispettivamente:

Monocolore:

BBB
BBB
BBB

A croce:

ABA
BBB
ABA

Ad anello:

BBB
BAB
BBB

A quattro angoli:

BAB
AAA
BAB

A cinque punti:

BAB
ABA
BAB

Risolvere il puzzle

Dopo aver scelto il tipo di variazione desiderata, lo schermo presenterà la griglia di lavoro e la griglia più piccola indicante l'obiettivo da raggiungere.

La posizione dei rettangoli blu e arancioni sulla griglia di lavoro viene scelta a caso dal computer: questa tecnica permette di avere più di 500 diverse combinazioni iniziali possibili.

Un cursore lampeggiante appare al centro della griglia: selezionate con il cursore il rettangolo che desiderate modificare e premete il pulsante di fuoco. Si noti che possono essere modificati soltanto i rettangoli arancioni: se premete il pul-

sante di fuoco mentre vi trovate su un rettangolo blu non accadrà nulla.

Supponendo che i rettangoli della griglia siano numerati nel modo seguente:

123
456
789

si avrà che:

-selezionando uno dei quattro rettangoli posti ai vertici della griglia (1, 3, 7, 9) si otterrà come risultato l'inversione del colore del rettangolo selezionato e dei tre rettangoli adiacenti.

-selezionando uno dei rettangoli posti al centro di ogni lato della griglia (2, 4, 6, 8) si otterrà invece l'inversione di colore del rettangolo selezionato e dei due rettangoli d'angolo adiacenti.

-selezionando infine il rettangolo centrale (5) si ottiene l'inversione del suo colore e di quello dei quattro rettangoli ai vertici della griglia.

Si noti che, utilizzando le variazioni 2-5, se si portano tutti i rettangoli sul colore blu non ci saranno più mosse possibili (come detto, non è possibile modificare il colore di un rettangolo blu) e il gioco avrà termine.

Match Blox

```

100 GOSUB 820:POKE 53280,6:POKE 53281,6:PR
    INT CHR$(8)"{CLR}"{ 2 GIU' }{<7>}"SPC(1
    4)"MATCH BLOX{ 2 GIU' }" :rem 206
110 L$(1)="{<1>}{RVS}{ 1 SH O }{<Y>}"
    { 1 SH P }{GIU' }{ 3 SIN }{<H>}"{<N>}"
    {GIU' }{ 3 SIN }{ 1 SH L }{<P>}"{ 1 SH @ }
    {OFF}"{ 2 SU }":L$(2)="{<7>}"+"RIGHT$(L$
    (1),21) :rem 236
120 NA$(1)="MONOCOLORE":NA$(2)="{
    { 2 SPAZI }CROCE":NA$(3)="ANELLO
    { 3 SPAZI }" :rem 80
130 NA$(4)="{ 4 ANGOLI }":NA$(5)="{ 5 PUNTI
    }":MV$="MOSSE" :rem 253
140 DIM Y(10),P$(10),CL(5,9):FOR I=1 TO 9
    :READ V(I,1),V(I,2),V(I,3),V(I,4)
    :rem 58
150 NEXT MT=1:UP=8:DN=14:LM=16:RM=22:TE=5
    :QX=11:QY=19:MV=-1:Q=14:X=8:P=0
    :rem 198
160 FOR C=1 TO 9:Y(C)=X:P$(C)=L$(2):Z=INT
    (10*RND(1)) :rem 55
170 IF Z<=5 THEN Y(C)=Q:P$(C)=L$(1)
    :rem 74
180 NEXT :rem 216
190 T$="{RVS}"{ 1 SH O }{ 1 SH P }{GIU' }
    { 2 SIN }{ 1 SH L }{ 1 SH @ }{SU}" :FOR R
    X=1 TO 5:FOR I=1 TO 9:CL(RX,I)=8:NEXT
    I,RX :rem 177
200 FOR I=1 TO 9:CL(1,I)=14:CL(3,I)=14:NE
    XT:CL(3,5)=8 :rem 110
210 CL(2,2)=14:CL(2,4)=14:CL(2,5)=14:CL(2
    ,6)=14:CL(2,8)=14 :rem 228
220 CL(4,1)=14:CL(4,3)=14:CL(4,7)=14:CL(4
    ,9)=14 :rem 158
230 CL(5,1)=14:CL(5,3)=14:CL(5,5)=14:CL(5
    ,7)=14:CL(5,9)=14 :rem 245
240 PRINT SPC(12)"{<8>}"{<P>}"{GIU' }
    { 2 SIN }{<N>}"{RVS}"{1 OFF}"{<H>}"{GIU' }
    { 2 SIN }{<Y>}"{SU}"{DES}"{<7>}"MONOCOLORE
    {GIU' }":PRINT SPC(12); :rem 189
250 PRINT "{<8>}"{<P>}"{GIU' }{ 2 SIN }{<N>}"
    {RVS}"{2 OFF}"{<H>}"{GIU' }{ 2 SIN }{<Y>}"
    {SU}"{DES}"{<7>}"CROCE{GIU' }":PRINT SPC(
    12)"{<8>}"{<P>}"{GIU' }{ 2 SIN }{<N>}"
    {RVS}"{3 OFF}"{<H>}"{GIU' }{ 2 SIN }{<Y>}"
    {SU}"{DES}"{<7>}"ANELLO{ 3 SPAZI }{GIU' }
    " :rem 125
260 PRINT SPC(12)"{<8>}"{<P>}"{GIU' }
    { 2 SIN }{<N>}"{RVS}"{4 OFF}"{<H>}"{GIU' }
    { 2 SIN }{<Y>}"{SU}"{DES}"{<7>}"4 ANGOLI
    {GIU' }" :rem 23
270 PRINT SPC(12)"{<8>}"{<P>}"{GIU' }
    { 2 SIN }{<N>}"{RVS}"{5 OFF}"{<H>}"{GIU' }
    { 2 SIN }{<Y>}"{SU}"{DES}"{<7>}"5 PUNTI
    { 3 GIU' }" :rem 18
280 PRINT TAB(12);"SCELTA{ 5 SPAZI }{SU}"
    { 2 SPAZI }{<8>}"{<P>}"{GIU' }{ 2 SIN }
    {<N>}"{RVS}"{OFF}"{<H>}"{GIU' }{ 2 SIN }
    {<Y>}"{<7>}" :rem 177
290 S=127-PEEK(56320):IF S=4 THEN MT=MT-1
    :IF MT<1 THEN MT=1 :rem 22
300 IF S=8 THEN MT=MT+1:IF MT>5 THEN MT=5
    :rem 19
310 IF S>=15 THEN DZ=MT:GK=MT:GOTO 330
    :rem 181
320 POKE 1929,MT+176:POKE 56201,15:GOTO 2
    90 :rem 54
330 PRINT CHR$(31)"{CLR}"; :rem 235
340 FOR I=0 TO 23:PRINT "{RVS}"{BLU}
    { 39 SPAZI }" :rem 120
350 MM=1063+(40*I):POKE MM,160:POKE MM+54
    272,6:NEXT:POKE 2023,160:POKE 56295,6
    :rem 40
360 PRINT "{RVS}"{ 39 SPAZI }{HOME}":POKE 53
    281,15 :rem 141
370 PRINT "{HOME}"{ 2 GIU' }";:PRINT SPC(15)
    "{RVS}"{NA$(GK)}{ 2 GIU' }" :rem 145
380 PRINT "{HOME}"{ 7 GIU' }{SPC(3)}{RVS}"
    {BLU}"{GIU' }A{GIU' }T{GIU' }C{GIU' }H
    { 2 GIU' }{ 3 SIN }B{GIU' }L{GIU' }O
    {GIU' }X" :rem 231
390 E=1:F=3:PRINT "{HOME}"{ 8 GIU' }":FOR BJ
    =1 TO 3:PRINT SPC(31);:FOR AI=E TO F
    :rem 137
400 POKE 646,CL(GK,AI):PRINT T$;:NEXT E=E
    +3:F=F+3:PRINT "{GIU' }":NEXT :rem 255
410 GOSUB 420:GOTO 450 :rem 183
420 PRINT "{HOME}"{ 7 GIU' }"TAB(15)"{RVS}"P
    $(1)P$(2)P$(3):PRINT "{ 2 GIU' }"TAB(15
    )" {RVS}"P$(4)P$(5)P$(6) :rem 185

```

Risolvere il puzzle

Dopo aver scelto il tipo di variazione desiderata, lo schermo presenterà la griglia di lavoro e la griglia più piccola indicante l'obiettivo da raggiungere.

La posizione dei rettangoli blu e arancioni sulla griglia di lavoro viene scelta a caso dal computer: questa tecnica permette di avere più di 500 diverse combinazioni iniziali possibili.

Un cursore lampeggiante appare al centro della griglia: selezionate con il cursore il rettangolo che desiderate modificare e premete il pulsante di fuoco. Si noti che possono essere modificati soltanto i rettangoli arancioni: se premete il pul-

sante di fuoco mentre vi trovate su un rettangolo blu non accadrà nulla. Supponendo che i rettangoli della griglia siano numerati nel modo seguente:

```

123
456
789

```

si avrà che:

-selezionando uno dei quattro rettangoli posti ai vertici della griglia (1, 3, 7, 9) si otterrà come risultato l'inversione del colore del rettangolo selezionato e dei tre rettangoli adiacenti.

-selezionando uno dei rettangoli posti al centro di ogni lato della griglia (2, 4, 6, 8) si otterrà invece l'inversione di colore del rettangolo selezionato e dei due rettangoli d'angolo affiancati.

-selezionando infine il rettangolo centrale (5) si ottiene l'inversione del suo colore e di quello dei quattro rettangoli ai vertici della griglia.

Si noti che, utilizzando le variazioni 2-5, se si portano tutti i rettangoli sul colore blu non ci saranno più mosse possibili (come detto, non è possibile modificare il colore di un rettangolo blu) e il gioco avrà termine.

Match Blox

```

100 GOSUB820:POKE 53280,6:POKE 53281,6:PR
    INT CHR$(8)"[CLR]"[2 GIU']"<7>"SPC(1
    4)"MATCH BLOX{2 GIU'}" :rem 206
110 L$(1)="[<1>]{RVS}{1 SH O}{<Y>}"
    {1 SH P}{GIU'}{3 SIN}{<H>}{<N>}"
    {GIU'}{3 SIN}{1 SH L}{<P>}{1 SH @}
    {OFF}{2 SU}":L$(2)="[<7>]+RIGHT$(L$
    (1),21) :rem 236
120 NA$(1)="MONOCOLORE":NA$(2)=""
    {2 SPAZI}CROCE":NA$(3)="ANELLO
    {3 SPAZI}" :rem 80
130 NA$(4)="4 ANGOLI ":NA$(5)="5 PUNTI "
    :MV$="MOSSE" :rem 253
140 DIM Y(10),P$(10),CL(5,9):FOR I=1 TO 9
    :READ V(I,1),V(I,2),V(I,3),V(I,4)
    :rem 58
150 NEXT MT=1:UP=8:DN=14:LM=16:RM=22:TE=5
    :QX=11:QY=19:MV=-1:Q=14:X=8:P=0
    :rem 198
160 FOR C=1 TO 9:Y(C)=X:P$(C)=L$(2):Z=INT
    (10*RND(1)) :rem 55
170 IF Z<=5 THEN Y(C)=Q:P$(C)=L$(1)
    :rem 74
180 NEXT :rem 216
190 T$="[RVS]{1 SH O}{1 SH P}{GIU'}"
    {2 SIN}{1 SH L}{1 SH @}{SU}":FOR R
    X=1 TO 5:FOR I=1 TO 9:CL(RX,I)=8:NEXT
    I,RX :rem 177
200 FOR I=1 TO 9:CL(1,I)=14:CL(3,I)=14:NE
    XT:CL(3,5)=8 :rem 110
210 CL(2,2)=14:CL(2,4)=14:CL(2,5)=14:CL(2
    ,6)=14:CL(2,8)=14 :rem 228
220 CL(4,1)=14:CL(4,3)=14:CL(4,7)=14:CL(4
    ,9)=14 :rem 158
230 CL(5,1)=14:CL(5,3)=14:CL(5,5)=14:CL(5
    ,7)=14:CL(5,9)=14 :rem 245
240 PRINT SPC(12)"[<8>][<P>]{GIU'}"
    {2 SIN}{<N>}{RVS}1{OFF}{<H>}{GIU'}"
    {2 SIN}{<Y>}{SU}{DES}{<7>}"MONOCOLORE
    {GIU'}":PRINT SPC(12); :rem 189
250 PRINT"[<8>][<P>]{GIU'}{2 SIN}{<N>}"
    {RVS}2{OFF}{<H>}{GIU'}{2 SIN}{<Y>}"
    {SU}{DES}{<7>}"CROCE{GIU'}":PRINTSPC(
    12)"[<8>][<P>]{GIU'}{2 SIN}{<N>}"
    {RVS}3{OFF}{<H>}{GIU'}{2 SIN}{<Y>}"
    {SU}{DES}{<7>}"ANELLO{3 SPAZI}{GIU'}"
    :rem 125
260 PRINT SPC(12)"[<8>][<P>]{GIU'}"
    {2 SIN}{<N>}{RVS}4{OFF}{<H>}{GIU'}"
    {2 SIN}{<Y>}{SU}{DES}{<7>}"4 ANGOLI
    {GIU'}" :rem 23
270 PRINT SPC(12)"[<8>][<P>]{GIU'}"
    {2 SIN}{<N>}{RVS}5{OFF}{<H>}{GIU'}"
    {2 SIN}{<Y>}{SU}{DES}{<7>}"5 PUNTI
    {3 GIU'}" :rem 18
280 PRINT TAB(12);"SCELTA{5 SPAZI}{SU}
    {2 SPAZI}{<8>}[<P>]{GIU'}{2 SIN}
    {<N>}{RVS}{OFF}{<H>}{GIU'}{2 SIN}
    {<Y>}[<7>]" :rem 177
290 S=127-PEEK(56320):IF S=4 THEN MT=MT-1
    :IF MT<1 THEN MT=1 :rem 22
300 IF S=8 THEN MT=MT+1:IF MT>5 THEN MT=5
    :rem 19
310 IF S>=15 THEN DZ=MT:GK=MT:GOTO 330
    :rem 181
320 POKE 1929,MT+176:POKE 56201,15:GOTO 2
    90 :rem 54
330 PRINT CHR$(31)"[CLR]"; :rem 235
340 FOR I=0 TO 23:PRINT"{RVS}{BLU}"
    {39 SPAZI}" :rem 120
350 MM=1063+(40*I):POKE MM,160:POKE MM+54
    272,6:NEXT:POKE 2023,160:POKE 56295,6
    :rem 40
360 PRINT"{RVS}{39 SPAZI}{HOME}":POKE 53
    281,15 :rem 141
370 PRINT"{HOME}{2 GIU'}":PRINT SPC(15)
    "{RVS}"NA$(GK)"{2 GIU'}" :rem 145
380 PRINT"{HOME}{7 GIU'}"SPC(3)"{RVS}"
    {BLU}M{GIU'}A{GIU'}T{GIU'}C{GIU'}H
    {2 GIU'}{3 SIN}B{GIU'}L{GIU'}O
    {GIU'}X" :rem 231
390 E=1:F=3:PRINT"{HOME}{8 GIU'}":FOR BJ
    =1 TO 3:PRINT SPC(31);:FOR AI=E TO F
    :rem 137
400 POKE 646,CL(GK,AI):PRINT T$;:NEXT:E=E
    +3:F=F+3:PRINT"{GIU'}":NEXT :rem 255
410 GOSUB 420:GOTO 450 :rem 183
420 PRINT"{HOME}{7 GIU'}"TAB(15)"{RVS}"P
    $(1)P$(2)P$(3):PRINT"{2 GIU'}"TAB(15)
    "{RVS}"P$(4)P$(5)P$(6) :rem 185

```

```

430 PRINT"{ 2 GIU' }"TAB(15)"{RVS}"P$(7)P$
      (8)P$(9) :rem 101
440 POKE 782,QY:POKE 781,QX:POKE 783,0:SY
      S 65520:PRINT"{BLK}[<+>]{SIN}";:RETUR
      N :rem 241
450 MV=MV+1:U=0:FOR C=1 TO 9:IF Y(C)=X TH
      EN U=U+1 :rem 223
460 NEXT:IF TC=1 THEN TC=0:MV=MV-1
      :rem 245
470 IF MV>0 THEN PRINT"{HOME}{ 20 GIU' }
      {RVS}{BLU}"SPC(16)"MOSSA"MV:GOSUB 440
      :rem 45
480 IF GK=1 AND U=9 THEN 750 :rem 148
490 IF U=9 THEN 780 :rem 197
500 IF GK=2 AND U=5 AND Y(2)+Y(4)+Y(5)+Y(
      6)+Y(8)=40 THEN 750 :rem 5
510 IF GK=3 AND U=8 AND Y(5)=14 THEN 750
      :rem 227
520 IF GK=4 AND U=4 AND Y(1)+Y(3)+Y(7)+Y(
      9)=32 THEN 750 :rem 255
530 IF GK=5 AND U=5 AND Y(1)+Y(3)+Y(5)+Y(
      7)+Y(9)=40 THEN 750 :rem 11
540 GOTO 610 :rem 105
550 P=U:IF Y(U)=X THEN TC=1:GOSUB 420:GOT
      O 450 :rem 105
560 W(1)=V(U,1):W(2)=V(U,2):W(3)=V(U,3):W
      (4)=V(U,4) :rem 17
570 FOR G=1 TO 4:W=0:IF Y(W(G))=X THEN W=
      1 :rem 248
580 IF W=1 THEN Y(W(G))=Q:P$(W(G))=L$(1)
      :rem 100
590 IF W=0 AND Y(W(G))=Q THEN Y(W(G))=X:P
      $(W(G))=L$(2) :rem 102
600 NEXT:Y(U)=X:P$(U)=L$(2):GOSUB 420:GOTO
      450 :rem 131
610 POKE 204,0:JY=15-PEEK(56320)AND15:JB=
      16-PEEK(56320)AND16 :rem 91
620 IF JB=0 THEN GOSUB 820:U=TE:GOTO 550
      :rem 219
630 IF JY=8 THEN JY=3 :rem 130
640 IF JY<1 OR JY>4 THEN 610 :rem 181
650 GOSUB 820:ON JY GOTO 660,680,700,720
      :rem 86
660 IF QX=UP THEN 610 :rem 124
670 QX=QX-3:TE=TE-3:GOTO 740 :rem 163
680 IF QX=DN THEN 610 :rem 107
690 QX=QX+3:TE=TE+3:GOTO 740 :rem 161
700 IF QY=RM THEN 610 :rem 114
710 QY=QY+3:TE=TE+1:GOTO 740 :rem 154
720 IF QY=LM THEN 610 :rem 110
730 QY=QY-3:TE=TE-1 :rem 146
740 GOSUB 420:GOTO 610 :rem 187
750 IF MV=1 THEN MV$="MOSSA" :rem 54
760 GOSUB 830:PRINT"{RVS}{BLU}{ 6 SPAZI }
      LO HAI RISOLTO IN"MV;MV$:S1=30:S2=20:
      S3=-1 :rem 224
770 GOSUB 840:GOTO 800 :rem 197
780 GOSUB 830:PRINT"{RVS}{BLU} NESSUNA MO
      SSA CONSENTITA:GAME OVER":S1=17:S2=35
      :S3=1 :rem 19
790 GOSUB 840 :rem 188
800 PRINT"{GIU' }{RVS}{ 5 SPAZI }PREMI FUOC
      O PER GIOCARE ANCORA{HOME}" :rem 45
510 WAIT 56320,16,16:FOR I=54272 TO 54287
      :POKE I,0:NEXT:RUN :rem 2
820 POKE 204,1:POKE 207,0:RETURN:rem 52
830 GOSUB 440:PRINT"{RVS}[<7>] {HOME}
      { 19 GIU' }":RETURN :rem 218
840 POKE 54296,15:POKE 54295,0:POKE 54273
      ,40:POKE 54278,240:POKE 54277,0
      :rem 54
850 POKE 54276,17:FOR A=S1 TO S2 STEP S3:
      FOR B=A TO 60-A :rem 205
860 POKE 54273,B:NEXT B,A:POKE 54273,3:PO
      KE 54276,0:RETURN :rem 28
870 DATA 2,4,5,10,1,3,10,10,2,5,6,10,1,7,
      10,10,2,4,8,6,3,9,10,10,4,5,8,10,7,9
      :rem 115
880 DATA 10,10,8,5,6,10,0,0,0,0,0,0,0,0,0,
      0,0,0,0,0,0,1,240,0,3,248,0,7,252,0
      :rem 68
890 DATA 7,252,0,7,252,0,3,248,0,1,240,0,
      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      :rem 15
900 DATA 0,0,0,0,0,0,0,0,0,0,190 :rem 137

```

NOVITA' IN EDICOLA



GIOCHI PARLANTI

Per la prima volta, una raccolta di giochi che utilizzano la sintesi vocale per dialogare con l'utente.

La Tombola, il Poker e il gioco del Sette e mezzo nella simpatica versione gestita dal computer.

Piccola guida per l'input dei programmi

I programmi in BASIC listati su **SUPER-COMMODORE** contengono una particolare codifica dei caratteri di controllo (cursore, colore, reverse, eccetera) che permette di ottenere una maggiore leggibilità dei programmi.

Generalmente, questi listati contengono alcuni caratteri racchiusi tra parentesi graffe { }; queste ultime, che non esistono sulla tastiera del computer, non devono essere digitate, ma hanno unicamente lo scopo di indicare che i caratteri da esse racchiusi sono dei caratteri di controllo.

Ad esempio, {GIU'} indica che occorre premere una volta il tasto di cursore verso il basso, {3 GIU'} indica che il tasto di cursore verso il basso dovrà essere premuto tre volte.

Se tra le parentesi graffe è racchiuso un singolo carattere, quest'ultimo deve essere premuto insieme al tasto CTRL (ad esempio, incontrando {A} si dovrà premere il tasto CTRL

insieme al tasto A).

Invece i caratteri racchiusi tra parentesi quadre e simboli di maggiore e minore [< >] devono essere premuti unitamente al tasto Commodore (nell'angolo inferiore sinistro della tastiera). Ad esempio, [<A>] indica che deve essere premuto il tasto Commodore insieme al tasto A.

Infine, alcuni caratteri racchiusi dalle parentesi graffe preceduti dalle lettere SH: ciò indica che il carattere seguente deve essere digitato tenendo premuto il tasto SHIFT.

La tabella seguente riporta, per ogni carattere di controllo, la codifica utilizzata nei nostri listati.

I listati in linguaggio macchina (totalmente numerici) non possono invece essere copiati direttamente con il computer, ma richiedono l'utilizzo dello speciale programma MLX, riportato, insieme alle istruzioni per il suo utilizzo, in altra parte della rivista.

Quando leggete:	Premete:	Vedrete:	Quando leggete:	Premete:	Vedrete:	Quando leggete:	Premete:	Vedrete:
{CLR}	SHIFT CLR/HOME		{CYN}	CTRL 4		[<7>]	G 7	
{HOME}	CLR/HOME		{PUR}	CTRL 5		[<8>]	G 8	
{SU}	SHIFT CSR		{GRN}	CTRL 6		{F1}	F1	
{GIU'}	CSR		{BLU}	CTRL 7		{F2}	F2	
{SIN}	SHIFT <= CSR =>		{YEL}	CTRL 8		{F3}	F3	
{DES}	<= CSR =>		{<1>}	G 1		{F4}	F4	
{RVS}	CTRL 9		{<2>}	G 2		{F5}	F5	
{OFF}	CTRL 0		{<3>}	G 3		{F6}	F6	
{BLK}	CTRL 1		{<4>}	G 4		{F7}	F7	
{WHT}	CTRL 2		{<5>}	G 5		{F8}	F8	
{RED}	CTRL 3		{<6>}	G 6				

PLAY

Q-Bird

di M. Sedore

trad. e adatt. di S. Colombo

Ecco un gioco di stile "arcade" dotato di ottime caratteristiche grafiche e di elevata velocità. E' richiesto l'utilizzo di un joystick.

Q-Bird viene giocato su una scacchiera composta da 6 quadrati per 7. Esistono 15 livelli di gioco tra cui scegliere, mentre il passaggio dal livello più semplice a quello più difficoltoso è automatico durante lo svolgimento del gioco.

Scopo del gioco è colorare tutti i quadrati costituenti la griglia con il colore mostrato nella parte bassa dello schermo. Per modificare il colore di un quadrato è sufficiente saltarvi sopra, prestando tuttavia la massima attenzione a non entrare in collisione con le altre creature che si muovono sulla griglia.

Nel livello 1 il colore di un quadrato viene modificato saltando sopra al quadrato stesso una volta, mentre nel livello 2 occorrerà saltarci sopra due volte, tre nel livello 3, e così via...

Si otterrà una vita in bonus al superamento di ogni livello, mentre il gioco inizia con cinque vite a disposizione.

Il livello iniziale di gioco può essere selezionato muovendo in avanti o indietro la leva del joystick (collegato in Porta 2) e premendo quindi il pulsante di fuoco.

Per saltare è sufficiente muovere la leva del joystick nella direzione in cui si desidera andare; premendo anche il pulsante di fuoco si otterrà un "super salto" della lunghezza di due quadrati. Occor-

re prestare attenzione a non entrare in contatto con le creature nemiche e a non cadere dalla griglia: in entrambi questi casi si perde infatti una vita. Un aiuto è rappresentato da un uovo lampeggiante che a volte appare sulla griglia: se ci saltate sopra, tutti i nemici verranno momentaneamente eliminati,

dandovi un po' di respiro nella tensione del gioco. Quando il vostro Q-Bird inizia a lampeggiare il tempo di scomparsa dei nemici sta per scadere, ed essi ricompariranno minacciosi in corrispondenza del terzo lampeggio.

Il gioco può essere "congelato" in qualsiasi momento tenendo premuto il tasto



SHIFT, oppure reinizializzato premendo RUN/STOP e RESTORE.

Prima di caricare e attivare MLX occorre tuttavia digitare questa serie di istruzioni:

indirizzo iniziale: 2049
indirizzo finale: 6568

Note per la digitazione di Q-Bird

Coloro che devono digitare il programma copiando il listato dalla rivista, dovranno come sempre servirsi di MLX (vedi in altra parte della rivista).

POKE 44,26:POKE 6656,0:NEW

Si dovranno quindi fornire a MLX i seguenti indirizzi:

Una volta salvato il programma su nastro o disco, esso potrà essere caricato e attivato come un qualsiasi programma BASIC.

Q-Bird

2049 :011,008,000,000,158,050,228
2055 :048,054,049,000,000,000,158
2061 :169,000,141,032,208,032,083
2067 :207,020,032,027,008,076,133
2073 :199,011,169,147,032,210,025
2079 :255,169,008,032,210,255,192
2085 :169,142,032,210,255,169,246
2091 :004,133,002,169,014,056,165
2097 :229,002,234,234,133,211,068
2103 :165,002,133,214,032,016,105
2109 :229,160,006,162,014,189,053
2115 :099,008,032,210,255,202,105
2121 :016,247,136,016,242,169,131
2127 :013,032,210,255,032,210,063
2133 :255,230,002,230,002,165,201
2139 :002,201,016,208,206,076,032
2145 :115,008,029,145,183,183,248
2151 :152,157,157,157,017,169,144
2157 :146,032,169,018,005,000,223
2163 :160,009,169,000,133,002,076
2169 :166,002,189,176,008,201,095
2175 :255,240,008,032,210,255,103
2181 :230,002,076,121,008,136,194
2187 :016,003,076,198,019,232,171
2193 :230,002,189,176,008,133,115
2199 :211,232,230,002,189,176,167
2205 :008,133,214,072,138,072,026
2211 :152,072,032,016,229,104,000
2217 :168,104,170,104,076,133,156
2223 :008,255,033,005,028,031,023
2229 :031,031,031,031,031,255,079
2235 :033,006,031,031,031,031,094
2241 :031,031,255,035,008,031,072
2247 :031,031,255,034,010,255,047
2253 :033,012,031,145,145,145,204
2259 :081,045,066,073,082,068,114
2265 :255,004,017,030,072,073,156
2271 :083,067,079,082,069,058,149
2277 :153,048,048,048,048,048,110
2283 :255,005,019,030,083,067,182
2289 :079,082,069,058,153,048,218
2295 :048,048,048,048,255,025,207
2301 :017,030,076,069,086,069,088
2307 :076,058,153,048,049,255,130
2313 :025,019,030,076,073,086,062
2319 :069,083,058,153,255,169,034
2325 :151,032,210,255,160,002,063

2331 :169,164,032,049,009,160,098
2337 :016,169,163,032,049,009,215
2343 :160,020,169,164,032,049,121
2349 :009,076,069,009,132,214,042
2355 :160,000,132,211,072,032,146
2361 :016,229,104,162,038,032,126
2367 :210,255,202,016,250,096,068
2373 :169,031,133,211,169,003,017
2379 :133,214,032,016,229,162,093
2385 :012,169,167,032,210,255,158
2391 :169,157,032,210,255,169,055
2397 :017,032,210,255,202,016,057
2403 :238,096,000,000,169,255,089
2409 :141,028,208,169,000,141,024
2415 :021,208,141,029,208,141,091
2421 :023,208,169,007,141,037,190
2427 :208,076,128,011,000,003,037
2433 :192,000,014,176,240,058,041
2439 :175,172,054,106,172,214,004
2445 :170,176,058,170,176,014,137
2451 :170,192,003,103,000,003,106
2457 :119,000,013,221,192,003,189
2463 :051,000,000,000,000,000,210
2469 :000,000,000,000,000,000,165
2475 :000,000,000,000,000,000,171
2481 :000,000,000,000,000,000,177
2487 :000,000,000,000,000,000,183
2493 :000,000,000,003,195,000,131
2499 :014,190,240,058,170,236,079
2505 :214,107,172,054,171,176,071
2511 :218,174,176,062,170,192,175
2517 :003,103,000,003,119,000,185
2523 :003,119,000,013,221,192,255
2529 :003,051,000,000,000,000,023
2535 :000,000,000,000,000,000,231
2541 :000,000,000,000,000,000,237
2547 :000,000,000,000,000,000,243
2553 :000,000,000,000,000,000,249
2559 :000,000,015,000,060,058,132
2565 :192,235,234,176,234,166,218
2571 :112,058,170,092,058,170,159
2577 :176,014,170,192,003,103,163
2583 :000,003,119,000,013,221,123
2589 :192,003,051,000,000,000,019
2595 :000,000,000,000,000,000,035
2601 :000,000,000,000,000,000,041
2607 :000,000,000,000,000,000,047

2613 :000,000,000,000,000,000,053
 2619 :000,000,000,000,000,003,062
 2625 :015,000,062,250,192,238,054
 2631 :170,176,235,166,092,059,201
 2637 :170,112,058,234,156,014,053
 2643 :170,240,003,103,000,003,090
 2649 :119,000,003,119,000,013,087
 2655 :221,192,003,051,000,000,050
 2661 :000,000,000,000,000,000,101
 2667 :000,000,000,000,000,000,107
 2673 :000,000,000,000,000,000,113
 2679 :000,000,000,000,000,000,119
 2685 :000,000,000,000,003,192,000,064
 2691 :014,176,000,054,108,000,227
 2697 :234,172,000,254,172,000,201
 2703 :234,172,000,062,176,000,019
 2709 :234,172,000,053,172,000,012
 2715 :013,172,000,013,172,000,013
 2721 :013,172,000,058,172,000,064
 2727 :234,176,000,062,176,000,047
 2733 :014,176,000,003,172,000,026
 2739 :000,240,000,000,000,000,163
 2745 :000,000,000,000,000,000,185
 2751 :000,003,192,000,014,176,064
 2757 :000,054,108,000,234,172,253
 2763 :000,234,172,000,058,176,075
 2769 :000,014,172,000,058,172,113
 2775 :000,229,172,000,061,172,081
 2781 :000,013,172,000,013,172,079
 2787 :000,058,176,000,234,176,103
 2793 :000,238,176,000,062,176,117
 2799 :000,003,176,000,000,192,098
 2805 :000,000,000,000,000,000,245
 2811 :000,000,000,000,000,015,010
 2817 :000,000,058,192,000,230,225
 2823 :112,000,234,172,000,234,247
 2829 :252,000,234,172,000,058,217
 2835 :240,000,234,172,000,233,130
 2841 :112,000,233,192,000,233,027
 2847 :192,000,233,192,000,234,114
 2853 :176,000,058,172,000,058,245
 2859 :240,000,058,192,000,235,000
 2865 :000,000,060,000,000,000,109
 2871 :000,000,000,000,000,000,055
 2877 :000,000,000,015,000,000,076
 2883 :058,192,000,230,112,000,147
 2889 :234,172,000,234,172,000,117
 2895 :058,176,000,234,192,000,227
 2901 :234,176,000,233,108,000,068
 2907 :233,240,000,233,192,000,221
 2913 :233,192,000,058,176,000,244
 2919 :058,172,000,058,236,000,115
 2925 :058,240,000,059,000,000,210
 2931 :012,000,000,000,000,000,127
 2937 :000,000,000,000,000,000,121
 2943 :000,169,000,141,038,208,171
 2949 :162,007,189,153,011,157,044
 2955 :039,208,189,161,011,157,136
 2961 :248,007,202,016,241,076,167

2967 :169,011,005,004,004,004,092
 2973 :009,002,006,014,128,044,104
 2979 :042,044,062,060,038,130,027
 2985 :162,015,189,183,011,157,118
 2991 :000,208,202,016,247,096,176
 2997 :000,000,120,102,109,071,071
 3003 :253,071,029,151,000,000,179
 3009 :000,000,169,158,000,000,008
 3015 :032,103,009,076,160,018,085
 3021 :076,085,025,173,052,003,107
 3027 :201,007,208,015,169,000,043
 3033 :141,052,003,238,053,003,195
 3039 :173,053,003,201,005,240,130
 3045 :003,076,008,022,032,058,172
 3051 :012,201,000,240,003,076,255
 3057 :196,017,032,219,017,041,251
 3063 :015,073,015,133,003,041,015
 3069 :001,201,001,208,008,169,073
 3075 :000,141,102,012,076,151,229
 3081 :012,165,003,041,002,201,177
 3087 :002,208,008,169,032,141,063
 3093 :102,012,076,151,012,165,027
 3099 :003,041,004,201,004,208,232
 3105 :008,169,064,141,102,012,017
 3111 :076,151,012,165,003,041,231
 3117 :008,201,008,208,180,169,051
 3123 :096,141,102,012,076,151,117
 3129 :012,169,000,141,053,003,179
 3135 :173,054,003,096,141,033,051
 3141 :208,169,000,162,203,157,200
 3147 :052,003,202,224,255,208,251
 3153 :248,096,238,054,003,174,126
 3159 :054,003,224,026,208,008,098
 3165 :169,000,141,054,003,076,024
 3171 :070,022,189,032,013,170,083
 3177 :189,128,012,024,109,012,067
 3183 :208,141,012,208,189,139,240
 3189 :012,024,109,013,208,141,112
 3195 :013,208,076,004,024,000,192
 3201 :000,000,000,255,255,255,126
 3207 :000,001,001,001,000,255,137
 3213 :001,000,000,255,001,000,142
 3219 :000,255,001,000,173,102,166
 3225 :012,201,000,240,012,201,051
 3231 :096,240,008,169,039,141,084
 3237 :254,007,076,083,012,169,254
 3243 :041,141,254,007,076,083,005
 3249 :012,206,254,007,076,230,194
 3255 :011,169,004,133,254,160,146
 3261 :000,132,253,162,004,173,145
 3267 :058,003,024,101,253,133,255
 3273 :253,165,254,105,000,133,087
 3279 :254,202,016,239,165,253,056
 3285 :024,109,059,003,133,253,026
 3291 :165,254,105,000,133,254,106
 3297 :096,000,000,173,013,208,203
 3303 :056,233,049,141,058,003,003
 3309 :173,012,208,074,074,074,084
 3315 :024,105,012,141,059,003,075

3321	:032,184,012,076,153,013,207	3675	:201,096,240,006,169,043,078
3327	:000,000,001,001,001,001,003	3681	:157,248,007,096,169,045,051
3333	:001,001,001,001,009,009,027	3687	:157,248,007,096,000,000,099
3339	:009,009,009,009,009,009,065	3693	:185,096,013,201,000,208,044
3345	:008,008,008,008,008,008,065	3699	:010,072,189,248,007,041,170
3351	:008,008,000,000,000,000,039	3705	:254,157,248,007,104,096,219
3357	:000,000,000,000,004,004,037	3711	:000,000,234,234,234,032,093
3363	:004,004,004,004,004,004,059	3717	:220,014,173,058,003,056,145
3369	:006,006,006,006,006,006,077	3723	:233,044,141,058,003,173,023
3375	:006,006,002,002,002,002,067	3729	:059,003,074,074,074,024,197
3381	:002,002,002,002,000,000,061	3735	:105,012,141,059,003,134,093
3387	:000,000,000,000,000,000,059	3741	:252,032,184,012,166,252,031
3393	:005,005,005,005,005,005,095	3747	:160,000,177,253,201,233,163
3399	:005,005,004,004,004,004,097	3753	:208,217,173,255,003,096,097
3405	:004,004,004,004,006,006,105	3759	:000,134,005,032,213,023,070
3411	:006,006,006,006,006,006,119	3765	:041,003,141,255,003,170,026
3417	:000,000,000,000,000,000,089	3771	:189,212,014,024,121,001,236
3423	:000,000,009,009,009,009,131	3777	:208,141,058,003,189,216,240
3429	:009,009,009,009,008,008,153	3783	:014,024,121,000,208,141,195
3435	:008,008,008,008,008,008,155	3789	:059,003,166,005,096,000,022
3441	:010,010,010,010,010,010,173	3795	:000,242,018,002,002,018,237
3447	:010,010,000,000,000,000,139	3801	:242,234,026,134,005,006,096
3453	:000,000,000,165,254,024,056	3807	:005,164,005,076,176,014,151
3459	:105,212,133,254,160,002,229	3813	:232,224,004,240,003,076,240
3465	:177,253,076,104,018,032,029	3819	:194,013,076,216,013,162,141
3471	:048,023,145,253,136,016,252	3825	:191,189,000,016,157,000,026
3477	:243,076,178,012,177,253,064	3831	:032,202,224,255,208,245,133
3483	:201,233,240,225,076,120,226	3837	:076,226,013,000,252,000,052
3489	:020,238,055,003,173,055,193	3843	:003,171,000,014,090,192,217
3495	:003,201,002,208,044,169,026	3849	:057,170,176,058,170,176,048
3501	:000,141,055,003,238,056,154	3855	:234,169,172,234,170,172,142
3507	:003,173,056,003,201,001,104	3861	:234,170,172,057,170,176,232
3513	:208,029,169,000,141,056,020	3867	:057,105,176,014,086,192,145
3519	:003,162,001,254,060,003,162	3873	:003,171,000,000,252,000,203
3525	:189,060,003,221,219,013,134	3879	:000,000,000,000,000,000,039
3531	:208,008,169,000,157,060,037	3885	:000,000,000,000,000,000,045
3537	:003,032,000,014,076,229,051	3891	:000,000,000,000,000,000,051
3543	:014,096,046,017,024,024,180	3897	:000,000,000,000,000,000,057
3549	:028,032,000,000,000,162,187	3903	:000,000,252,000,003,171,233
3555	:024,169,000,157,000,212,021	3909	:000,014,150,192,058,169,140
3561	:202,016,250,169,255,141,242	3915	:176,058,170,176,234,170,035
3567	:015,212,169,128,141,018,154	3921	:172,234,170,172,234,170,209
3573	:212,169,000,076,174,023,131	3927	:172,057,170,176,057,106,057
3579	:000,000,000,000,000,189,184	3933	:176,014,090,192,003,171,227
3585	:064,003,201,000,208,013,234	3939	:000,000,252,000,000,000,095
3591	:032,129,014,041,003,010,236	3945	:000,000,000,000,000,000,105
3597	:010,010,010,010,032,084,169	3951	:000,000,000,000,000,000,111
3603	:014,189,080,003,141,110,044	3957	:000,000,000,000,000,000,117
3609	:014,254,064,003,188,064,100	3963	:000,000,000,000,000,000,123
3615	:003,192,030,208,009,169,130	3969	:000,000,000,000,000,000,129
3621	:000,157,064,003,234,234,217	3975	:000,000,003,243,252,062,183
3627	:234,096,134,005,006,005,011	3981	:174,171,239,250,101,234,030
3633	:142,079,003,234,032,109,136	3987	:170,169,234,234,175,063,168
3639	:014,170,164,005,189,128,213	3993	:170,240,058,175,000,015,043
3645	:012,024,121,000,208,153,067	3999	:240,000,000,000,000,000,143
3651	:000,208,189,139,012,024,127	4005	:000,000,000,000,000,000,165
3657	:121,001,208,153,001,208,253	4011	:000,000,000,000,000,000,171
3663	:174,079,003,096,000,157,076	4017	:000,000,000,000,000,000,177
3669	:080,003,201,000,240,010,107	4023	:000,000,000,000,000,000,183

4029	:000,000,000,000,000,000,189	4383	:208,141,001,208,076,105,002
4035	:000,000,000,015,192,000,146	4389	:017,169,000,141,240,003,095
4041	:058,188,000,014,171,252,116	4395	:076,035,017,238,241,003,141
4047	:063,174,171,239,250,101,181	4401	:173,241,003,201,010,208,117
4053	:234,170,169,234,234,175,149	4407	:235,169,000,141,241,003,076
4059	:063,170,240,014,171,000,109	4413	:238,242,003,173,242,003,194
4065	:014,172,000,058,240,000,197	4419	:201,005,208,220,169,000,102
4071	:015,000,000,000,000,000,246	4425	:141,242,003,076,232,016,015
4077	:000,000,000,000,000,000,237	4431	:169,129,141,248,007,076,081
4083	:000,000,000,000,000,000,243	4437	:192,016,173,012,208,141,059
4089	:000,000,000,000,000,000,249	4443	:245,003,173,013,208,141,106
4095	:000,000,000,000,000,252,251	4449	:246,003,173,000,220,096,067
4101	:000,003,171,000,014,102,039	4455	:000,000,238,250,003,173,255
4107	:192,062,170,240,053,169,129	4461	:250,003,201,006,208,015,024
4113	:112,053,153,112,013,085,033	4467	:169,000,141,250,003,238,148
4119	:192,003,087,000,003,155,207	4473	:251,003,173,251,003,201,235
4125	:000,003,171,000,014,108,069	4479	:005,240,040,076,237,017,230
4131	:000,014,175,192,057,190,151	4485	:169,000,141,251,003,238,167
4137	:176,058,186,192,057,190,132	4491	:008,208,173,008,208,201,177
4143	:176,014,170,176,014,106,191	4497	:001,208,238,032,213,023,092
4149	:192,003,171,000,000,252,159	4503	:041,007,201,006,176,247,061
4155	:000,000,000,000,000,003,062	4509	:010,010,010,010,024,105,070
4161	:255,000,014,102,192,058,174	4515	:076,141,009,208,076,130,035
4167	:170,176,053,153,112,053,020	4521	:017,238,252,003,173,252,080
4173	:085,112,013,085,192,003,055	4527	:003,201,032,208,209,173,233
4179	:087,000,003,155,000,003,075	4533	:252,007,073,001,141,252,139
4185	:155,000,003,171,000,003,165	4539	:007,169,000,141,252,003,247
4191	:155,000,014,172,000,014,194	4545	:076,133,017,173,254,003,081
4197	:108,000,014,172,000,014,153	4551	:201,001,240,003,076,083,035
4203	:108,000,014,172,000,003,148	4557	:012,238,253,003,173,253,113
4209	:107,000,003,171,000,003,141	4563	:003,041,001,240,243,076,047
4215	:155,000,000,234,192,000,188	4569	:086,012,169,000,141,254,111
4221	:063,000,000,000,000,000,188	4575	:003,173,000,220,041,016,164
4227	:000,000,000,000,048,000,179	4581	:208,003,238,254,003,076,243
4233	:000,236,000,003,171,000,035	4587	:087,017,238,243,003,173,228
4239	:003,107,000,003,107,000,107	4593	:243,003,201,005,208,015,148
4245	:003,091,000,003,155,000,145	4599	:169,000,141,243,003,238,017
4251	:000,252,000,000,000,000,151	4605	:244,003,173,244,003,201,097
4257	:000,000,000,000,000,000,161	4611	:010,240,003,076,205,011,036
4263	:000,000,000,000,000,000,167	4617	:169,000,141,244,003,238,036
4269	:000,000,000,000,000,000,173	4623	:011,208,076,061,018,201,078
4275	:000,000,000,000,000,000,179	4629	:001,240,010,201,072,144,177
4281	:000,000,000,000,000,000,185	4635	:003,032,076,018,076,006,238
4287	:000,173,001,208,024,105,190	4641	:018,032,213,023,041,007,111
4293	:008,056,237,246,003,240,219	4647	:201,007,176,247,170,169,241
4299	:008,144,003,169,000,096,111	4653	:000,024,105,024,202,016,160
4305	:169,032,096,173,000,208,119	4659	:250,024,105,088,141,010,157
4311	:024,105,002,056,237,245,116	4665	:208,076,028,018,173,011,059
4317	:003,144,003,169,064,096,188	4671	:208,201,168,208,208,169,201
4323	:169,096,096,000,000,173,249	4677	:001,141,011,208,076,034,028
4329	:240,003,201,000,208,006,123	4683	:018,206,010,208,238,239,226
4335	:032,079,017,141,000,017,013	4689	:003,173,239,003,201,008,196
4341	:238,240,003,174,240,003,119	4695	:208,013,173,253,007,073,046
4347	:224,032,240,039,189,032,239	4701	:001,141,253,007,169,000,152
4353	:013,170,201,000,208,008,089	4707	:141,239,003,096,000,041,107
4359	:173,248,007,041,128,141,233	4713	:015,201,002,208,003,076,098
4365	:248,007,189,128,012,024,109	4719	:150,013,133,004,230,004,133
4371	:109,000,208,141,000,208,173	4725	:201,001,234,208,013,238,244
4377	:189,139,012,024,109,001,243	4731	:208,003,173,208,003,201,151

4737 :126,208,003,076,138,018,186
 4743 :076,142,013,165,004,145,168
 4749 :253,076,071,023,173,225,194
 4755 :003,141,118,018,141,107,163
 4761 :018,076,183,019,096,000,033
 4767 :000,169,006,133,211,169,079
 4773 :022,133,214,032,016,229,043
 4779 :162,000,189,187,018,032,247
 4785 :210,255,232,224,061,208,087
 4791 :245,076,106,021,008,142,013
 4797 :158,085,083,069,032,083,187
 4803 :084,073,067,075,032,084,098
 4809 :079,032,083,069,076,069,097
 4815 :067,084,032,076,069,086,109
 4821 :069,076,013,032,000,000,147
 4827 :000,032,032,032,032,032,123
 4833 :032,080,082,069,083,083,142
 4839 :032,084,082,073,071,071,132
 4845 :069,082,032,084,079,032,103
 4851 :083,084,065,082,084,000,129
 4857 :000,032,119,021,041,016,222
 4863 :208,011,032,114,019,169,040
 4869 :064,141,021,208,076,091,094
 4875 :023,173,000,220,041,015,227
 4881 :073,015,041,003,201,001,095
 4887 :208,013,173,225,003,201,078
 4893 :015,240,218,238,225,003,200
 4899 :076,052,019,201,002,208,081
 4905 :208,173,225,003,201,002,085
 4911 :144,201,206,225,003,173,231
 4917 :225,003,201,010,176,014,170
 4923 :024,105,048,141,200,006,071
 4929 :169,048,141,199,006,076,192
 4935 :250,018,024,105,038,141,135
 4941 :200,006,169,049,141,199,073
 4947 :006,076,250,018,000,238,159
 4953 :227,003,208,251,238,228,220
 4959 :003,173,228,003,201,255,190
 4965 :208,246,169,008,141,228,077
 4971 :003,173,000,220,096,000,087
 4977 :000,162,039,169,032,157,160
 4983 :112,007,157,152,007,202,244
 4989 :016,247,096,000,000,032,004
 4995 :145,018,076,167,019,238,026
 5001 :227,003,208,251,238,228,012
 5007 :003,208,246,238,229,003,046
 5013 :173,229,003,201,002,208,197
 5019 :236,169,000,141,229,003,165
 5025 :141,021,208,076,211,019,069
 5031 :173,225,003,201,000,208,209
 5037 :003,238,225,003,032,145,051
 5043 :018,076,224,019,238,107,093
 5049 :018,201,015,208,005,169,033
 5055 :000,141,107,018,096,000,041
 5061 :000,169,048,032,210,255,143
 5067 :169,053,032,210,255,076,230
 5073 :020,009,032,027,008,173,222
 5079 :225,003,201,015,240,003,134
 5085 :238,225,003,169,096,141,069

5091 :070,019,141,084,019,032,080
 5097 :052,019,169,076,141,070,248
 5103 :019,141,084,019,032,023,045
 5109 :020,032,228,022,169,255,203
 5115 :141,021,208,173,225,003,254
 5121 :133,004,169,000,162,203,160
 5127 :157,052,003,202,224,255,132
 5133 :208,248,165,004,141,225,236
 5139 :003,076,145,021,169,014,191
 5145 :133,211,169,022,133,214,139
 5151 :032,035,021,162,000,032,057
 5157 :076,020,232,224,011,208,040
 5163 :248,169,096,141,161,019,109
 5169 :032,136,019,169,141,141,175
 5175 :161,019,032,220,020,076,071
 5181 :145,018,158,071,069,084,094
 5187 :032,082,069,065,068,089,216
 5193 :000,000,000,189,063,020,089
 5199 :076,210,255,238,212,003,049
 5205 :208,251,238,213,003,173,147
 5211 :213,003,201,003,208,241,192
 5217 :169,000,141,213,003,032,143
 5223 :058,024,173,013,208,201,012
 5229 :255,208,226,169,000,141,084
 5235 :027,208,076,157,020,169,004
 5241 :064,141,027,208,076,082,207
 5247 :020,141,021,208,173,208,130
 5253 :003,133,252,169,096,141,159
 5259 :020,020,032,224,019,169,111
 5265 :076,141,020,020,165,252,051
 5271 :141,208,003,076,205,011,027
 5277 :141,021,208,173,024,007,219
 5283 :201,048,208,021,173,023,069
 5289 :007,201,048,208,003,076,200
 5295 :041,021,169,057,141,024,116
 5301 :007,206,023,007,076,191,179
 5307 :020,206,024,007,032,141,105
 5313 :025,141,199,019,173,024,006
 5319 :007,141,204,019,076,131,009
 5325 :020,000,169,048,141,199,014
 5331 :019,169,053,141,204,019,048
 5337 :076,240,014,169,012,133,093
 5343 :211,169,022,133,214,032,236
 5349 :016,229,162,000,189,247,048
 5355 :020,032,210,255,232,224,184
 5361 :023,208,245,076,016,021,062
 5367 :005,067,072,065,078,071,093
 5373 :069,032,084,079,032,018,055
 5379 :233,032,146,233,017,157,053
 5385 :157,157,152,183,183,000,073
 5391 :000,173,225,003,024,105,033
 5397 :001,141,134,219,141,135,024
 5403 :219,141,136,219,096,000,070
 5409 :000,000,032,114,019,076,018
 5415 :016,229,032,114,019,169,106
 5421 :014,133,211,169,022,133,215
 5427 :214,032,016,229,162,000,192
 5433 :189,094,021,032,210,255,090
 5439 :232,224,010,208,245,238,196

5445	:215,003,208,251,238,216,176	5799	:056,237,012,208,201,005,118
5451	:003,208,246,238,217,003,222	5805	:176,018,189,001,208,024,021
5457	:173,217,003,201,005,208,120	5811	:125,201,022,056,237,013,065
5463	:236,162,000,154,076,013,216	5817	:208,201,005,176,003,076,086
5469	:008,159,071,065,077,069,030	5823	:166,023,202,202,016,219,251
5475	:032,079,086,069,082,000,191	5829	:076,162,013,004,011,255,206
5481	:000,169,014,141,021,208,146	5835	:010,255,010,255,010,005,236
5487	:169,096,141,216,013,076,054	5841	:005,003,004,169,096,141,115
5493	:250,018,032,162,013,238,062	5847	:161,019,032,050,024,169,158
5499	:236,003,208,248,238,237,013	5853	:141,141,161,019,076,120,111
5505	:003,173,237,003,201,009,243	5859	:020,238,204,019,173,204,061
5511	:208,238,169,000,141,237,104	5865	:019,201,058,208,008,169,128
5517	:003,076,093,019,169,076,065	5871	:048,141,204,019,238,199,064
5523	:141,216,013,169,016,056,246	5877	:019,076,128,011,169,000,136
5529	:237,225,003,074,141,174,239	5883	:141,202,003,238,203,003,017
5535	:021,238,174,021,076,205,126	5889	:096,000,000,238,193,003,019
5541	:011,000,238,193,003,173,015	5895	:173,193,003,201,004,208,021
5547	:193,003,201,008,208,246,006	5901	:008,169,000,141,193,003,015
5553	:169,000,141,193,003,238,153	5907	:076,205,011,076,008,022,161
5559	:052,003,076,208,011,032,053	5913	:162,003,254,003,007,189,131
5565	:068,025,041,248,024,105,188	5919	:003,007,201,058,208,008,004
5571	:006,141,013,080,032,213,168	5925	:169,048,157,003,007,202,111
5577	:023,041,248,024,105,001,131	5931	:016,238,076,077,023,192,153
5583	:141,012,080,032,228,012,200	5937	:000,208,014,132,252,164,051
5589	:160,000,177,253,201,233,213	5943	:004,032,025,023,136,192,211
5595	:208,223,173,012,080,141,032	5949	:001,208,248,164,252,165,075
5601	:014,208,173,013,080,141,086	5955	:004,096,000,000,032,048,247
5607	:015,208,169,076,141,252,068	5961	:023,076,136,019,162,004,237
5613	:012,076,249,021,169,096,092	5967	:189,003,007,157,246,008,177
5619	:141,252,012,076,059,022,037	5973	:202,016,247,076,112,023,249
5625	:169,208,141,239,012,141,135	5979	:162,004,169,048,157,003,122
5631	:230,012,076,205,011,000,021	5985	:007,202,016,250,076,130,010
5637	:000,000,000,173,197,003,122	5991	:019,162,002,032,027,023,112
5643	:201,000,240,003,076,108,127	5997	:096,000,000,162,000,056,167
5649	:022,238,202,003,173,202,089	6003	:189,180,006,253,003,007,241
5655	:003,201,112,208,022,032,089	6009	:048,009,233,001,016,019,191
5661	:249,022,173,203,003,201,112	6015	:232,224,005,208,238,162,172
5667	:000,208,012,173,015,208,139	6021	:004,189,003,007,157,180,161
5673	:201,000,240,008,169,000,147	6027	:006,157,230,008,202,016,246
5679	:141,015,208,076,152,022,149	6033	:244,096,000,000,173,054,200
5685	:032,241,021,076,050,022,239	6039	:003,201,006,144,007,201,201
5691	:169,080,141,239,012,141,073	6045	:021,176,003,076,193,022,136
5697	:230,012,076,188,021,173,253	6051	:076,212,022,173,254,003,135
5703	:012,208,056,237,014,208,038	6057	:208,234,076,212,022,169,066
5709	:201,000,208,024,173,013,184	6063	:048,141,186,023,162,000,223
5715	:208,056,237,015,208,201,240	6069	:173,027,212,157,000,080,062
5721	:000,208,013,032,104,023,213	6075	:232,208,247,238,186,023,041
5727	:169,127,141,197,003,169,133	6081	:173,186,023,201,080,208,040
5733	:000,141,015,208,076,228,001	6087	:235,169,048,141,238,023,029
5739	:012,238,198,003,173,198,161	6093	:169,000,141,237,023,076,083
5745	:003,208,033,238,199,003,029	6099	:242,023,173,237,023,024,165
5751	:173,199,003,201,022,208,157	6105	:105,001,141,237,023,173,129
5757	:023,169,000,141,199,003,148	6111	:238,023,105,000,201,079,101
5763	:238,200,003,032,122,025,239	6117	:208,002,169,048,141,238,011
5769	:201,016,208,008,169,000,227	6123	:023,173,071,048,096,000,134
5775	:141,200,003,141,197,003,060	6129	:000,169,225,141,020,212,240
5781	:076,004,023,238,046,208,232	6135	:169,033,141,018,212,169,221
5787	:234,234,234,162,010,189,194	6141	:015,141,024,212,076,042,251
5793	:000,208,024,125,200,022,228	6147	:024,174,054,003,189,016,207

6153 : 024, 141, 015, 212, 076, 230, 195
 6159 : 011, 000, 128, 124, 120, 116, 002
 6165 : 112, 110, 108, 106, 104, 102, 151
 6171 : 000, 000, 000, 000, 000, 000, 027
 6177 : 000, 000, 000, 000, 000, 000, 033
 6183 : 000, 000, 000, 169, 000, 141, 093
 6189 : 005, 212, 076, 076, 025, 169, 096
 6195 : 000, 141, 015, 212, 076, 136, 119
 6201 : 019, 238, 013, 208, 173, 013, 209
 6207 : 208, 073, 255, 141, 015, 212, 199
 6213 : 096, 000, 000, 169, 008, 141, 227
 6219 : 009, 212, 141, 010, 212, 169, 060
 6225 : 119, 141, 006, 212, 141, 013, 201
 6231 : 212, 169, 065, 141, 011, 212, 129
 6237 : 169, 076, 141, 020, 020, 169, 176
 6243 : 000, 141, 015, 212, 076, 016, 047
 6249 : 025, 173, 254, 047, 201, 001, 038
 6255 : 208, 027, 174, 255, 047, 189, 243
 6261 : 008, 025, 141, 008, 212, 169, 168
 6267 : 040, 141, 001, 212, 169, 129, 047
 6273 : 141, 004, 212, 169, 128, 141, 156
 6279 : 004, 212, 076, 005, 025, 201, 146
 6285 : 015, 208, 040, 169, 064, 141, 010
 6291 : 011, 212, 169, 065, 141, 011, 244
 6297 : 212, 169, 030, 141, 001, 212, 150
 6303 : 238, 255, 047, 174, 255, 047, 151
 6309 : 189, 008, 025, 141, 008, 212, 236
 6315 : 169, 129, 141, 004, 212, 169, 227
 6321 : 128, 141, 004, 212, 076, 005, 231
 6327 : 025, 201, 016, 208, 015, 174, 054
 6333 : 255, 047, 189, 008, 025, 024, 225
 6339 : 105, 002, 141, 008, 212, 076, 227
 6345 : 005, 025, 201, 024, 208, 035, 187
 6351 : 169, 064, 141, 011, 212, 169, 205
 6357 : 065, 141, 011, 212, 174, 255, 047

6363 : 047, 189, 008, 025, 024, 105, 105
 6369 : 004, 141, 008, 212, 169, 129, 120
 6375 : 141, 004, 212, 169, 128, 141, 002
 6381 : 004, 212, 076, 005, 025, 201, 248
 6387 : 030, 208, 015, 032, 033, 025, 074
 6393 : 173, 255, 047, 201, 008, 208, 117
 6399 : 005, 169, 000, 141, 255, 047, 104
 6405 : 076, 044, 025, 004, 008, 012, 174
 6411 : 016, 020, 016, 012, 008, 120, 203
 6417 : 169, 050, 141, 020, 003, 169, 057
 6423 : 025, 141, 021, 003, 088, 169, 214
 6429 : 000, 076, 067, 012, 238, 255, 165
 6435 : 047, 169, 000, 141, 254, 047, 181
 6441 : 096, 000, 000, 238, 254, 047, 164
 6447 : 076, 049, 234, 238, 253, 047, 176
 6453 : 173, 253, 047, 041, 001, 201, 001
 6459 : 001, 240, 003, 076, 049, 234, 150
 6465 : 076, 106, 024, 032, 213, 023, 027
 6471 : 201, 176, 176, 249, 096, 141, 086
 6477 : 254, 047, 141, 255, 047, 076, 129
 6483 : 101, 025, 173, 141, 002, 208, 221
 6489 : 003, 076, 167, 021, 169, 000, 013
 6495 : 141, 015, 212, 076, 085, 025, 137
 6501 : 169, 088, 141, 024, 003, 169, 183
 6507 : 021, 141, 025, 003, 169, 000, 210
 6513 : 141, 016, 208, 141, 027, 208, 086
 6519 : 076, 072, 024, 173, 200, 003, 155
 6525 : 201, 011, 144, 008, 173, 045, 195
 6531 : 208, 073, 008, 141, 045, 208, 046
 6537 : 173, 200, 003, 096, 169, 064, 074
 6543 : 141, 021, 208, 169, 169, 141, 224
 6549 : 012, 208, 169, 158, 141, 013, 082
 6555 : 208, 169, 038, 141, 254, 007, 204
 6561 : 173, 023, 007, 096, 000, 000, 204
 6567 : 000, 000, 000, 000, 000, 000, 167

SUPERCOMMODORE

CEDOLA DI ORDINAZIONE RIVISTE ARRETRATE CON CASSETTA ALLEGATA
 da compilare e spedire in busta chiusa a
 Gruppo Editoriale Jackson - Via Rosellini, 12 - 20124 Milano - Tel. 6880951/2/3/4/5

Vogliate inviarmi i numeri

di SUPERCOMMODORE con cassettaAnno

Al prezzo di L. 15.000 cad.

☐ Contanti allegati ☐ Assegno allegato n°

☐ Ho spedito l'importo a mezzo vaglia postale

☐ Ho versato l'importo sul ccp. n° 11666203 intestato Gruppo Editoriale Jackson - Milano

☐ Pagherò in contrassegno al postino al ricevimento

BUONO D'ORDINE PER RIVISTE ARRETRATE

Anche se l'ordine riguardasse la sola cassetta, questa verrà comunque inviata insieme alla rivista al prezzo indicato.

Nome

Cognome

Via

CittàC.A.P.Prov.

Se richiesta fattura:

Cod. F. e P. Iva

Data

Firma

Per i residenti all'estero - pagamento anticipato (vaglia o versamento su ns. ccp)

PLAY

Connect'em

di F. Chambers
trad. e adatt. di E. Stefani

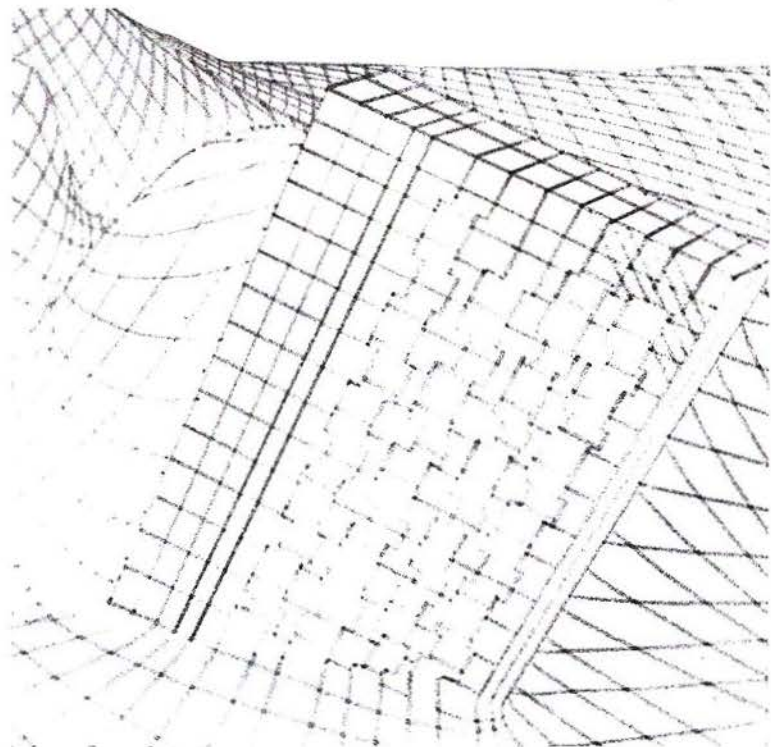
Un interessante gioco consistente nella versione informatica di un diffuso passatempo tipico delle riviste di enigmistica. E' possibile giocare contro il computer, contro un amico, oppure osservare il computer che gioca contro se E' richiesto l'uso di uno o due

Il gioco consiste nel collegare una serie di punti presenti su una scacchiera, cercando di chiudere il percorso con dei quadrati; una volta completato un quadrato, esso viene riempito con la propria iniziale ed il gioco prosegue nel tentativo di realizzare un nuovo quadrato. Quando tutti i quadrati possibili sono stati tracciati, viene dichiarato vincitore quello dei giocatori che ne ha totalizzato il maggior numero.

Il programma consente la scelta tra due diverse dimensioni della scacchiera: normale (composta da 84 quadrati) e mini (con 42 quadrati).

Dopo aver dato il RUN al programma, il computer chiede i nomi dei giocatori: se volete fare in modo che una (o entrambe) le posizioni siano giocate dal computer, digitate C-64 come nome del corrispondente giocatore.

Le mosse vengono effettuate con il joystick (occorrono due joystick se non si gioca contro il computer) posizionando il puntatore blu sul punto iniziale di tracciamento della linea, e premendo il pulsante di fuoco; occorrerà quindi portarsi con il puntatore sul punto di arrivo



della linea e premere nuovamente il pulsante di fuoco sul joystick. Il computer provvederà automaticamente al tracciamento della linea di congiunzione tra i due punti. Si noti che possono essere collegati tra loro soltanto punti adiacenti, e non sono permesse mosse che non

seguano questa regola.

Quando viene realizzato un quadrato, il computer provvede a riempirlo con l'iniziale appropriata del giocatore che lo ha realizzato, mentre al termine del gioco viene visualizzato il nome del giocatore vincente.

Connect 'Em

```

10 DIM BX(8,13),ZO(84),ZX(4,4) :rem 67
20 FORA=679TO743:READB:CH=CH+B:POKEA,B:NE
  XT :rem 209
30 FORX=0TO4:FORY=1TO4:READZX(X,Y):CH=CH+
  ZX(X,Y):NEXT:NEXT :rem 101
40 IFCH<>6391THENPRINT"ERRORE NEI DATI!":
  END :rem 202
50 POKE785,167:POKE786,2 :rem 110
60 DATA 32,247,183,165,20,201,3,176,26,73
  ,3,170,189,255,219,41,15,133,2,169
  :rem 111
70 DATA 15,56,229,2,170,189,221,2,168,169
  ,0,32,145,179,96,160,0,41,1,170,189
  :rem 167
80 DATA 0,220,41,16,208,1,200,169,0,32,14
  5,179,96,0,1,3,0,4,8,7,0,2,5,6
  :rem 118
90 DATA 1,40,42,81,-1,0,0,2,0,-1,0,80,0,1
  ,2,82,1,0,80,82 :rem 113
100 PRINT"{CLR}":S=1024:C=54272:CH=81:CL=
  2:OX=46:P1=0:P2=0:IN$="":LX=28:UX=12
  :rem 66
110 POKE53280,15:POKE53281,1:FORT=0TO3:PC
  $(T)="" :BZ$(T)="" :NEXT :rem 134
120 X=7:Y=6:V=S+40*X+Y:OV=V:PL=1:OL=46:DT
  $="" :rem 195
130 GOSUB710:IFPC$(1)<>"X"THENPOKEV,CH:PO
  KEV+C,14:X=7:Y=6:GOTO150 :rem 2
140 GOTO1210 :rem 146
150 A=USR(PL):IFA=0GOTO210 :rem 62
160 IFA=1THENX=X-1 :rem 79
170 IFA=2THENY=Y+1 :rem 81
180 IFA=3THENX=X+1 :rem 81
190 IFA=4THENY=Y-1 :rem 87
200 GOTO260 :rem 99
210 B=USR(PL+2):IFB=0GOTO150 :rem 157
220 IFOL=66GOTO690 :rem 63
230 IFOL<>46THENGOSUB1440:GOTO150
  :rem 36
240 GOSUB1460:IFD1=0THEND1=V:GOTO150
  :rem 204
250 D2=V:GOTO360 :rem 172
260 IFX<7THENX=7 :rem 237
270 IFX>21THENX=21 :rem 72
280 IFY<6THENY=6 :rem 239
290 IFY>LXTHENY=LX :rem 206
300 V=S+40*X+Y :rem 14
310 IFPC$(PL)="X"THENOL=OX:OC=OY:rem 255
320 POKEOV,OL:POKEOV+C,OC :rem 106
330 OL=PEEK(V):OC=PEEK(V+C):POKEV,CH:POKE
  V+C,14:OV=V:OX=OL :rem 220
340 IFPC$(PL)<"X"THENOY=OC :rem 72
350 GOTO150 :rem 103
360 AA=D1:BB=D2:BX$="" :IFD1<D2GOTO380
  :rem 123
370 A=D1:D1=D2:AA=D2:D2=A :rem 140
380 E=D2-D1:IFE=2THENDI$="A":GOTO410
  :rem 92
390 IFE=80THENDI$="D":GOTO470 :rem 200
400 GOSUB1440:D2=0:D1=0:GOTO150 :rem 31
410 E=PEEK(AA+1):IFE=111GOTO400 :rem 22

20 POKED1+1,111 :rem 85
430 IFPC$(PL)=""GOTO560 :rem 31
440 GOSUB1540 :rem 226
450 FORE=1TO8:POKED1+1,32:FORK=1TO50:NEXT
  :POKED1+1,111:FORK=1TO75:NEXT
  :rem 143
460 NEXT:GOTO560 :rem 231
470 E=PEEK(AA+40):IFE=66GOTO400 :rem 40
480 POKED1+40,66:POKED1+40+C,2:POKED2,66:
  POKED2+C,2:IFPC$(PC)="X"GOTO540
  :rem 209
490 IFPC$(PL)=""GOTO540 :rem 35
500 GOSUB1540:IFV=D2THENOX=66:OY=2
  :rem 129
510 FORE=1TO8:POKED1+40,32:POKED2,32:FORK
  =1TO50:NEXT:POKED1+40,66:POKED2,66
  :rem 80
520 POKED1+40+C,2:POKED2+C,2 :rem 65
530 FORK=1TO75:NEXT:NEXT :rem 57
540 IFOV=D2ANDPL=1THENOX=66:OY=2:rem 45
550 IFOV=D2THENOL=66:OC=2 :rem 47
560 XX=AA-S:FORK=1TO21:XX=XX-40:YY=XX:IFY
  Y<40THENXX=K:K=21 :rem 82
570 NEXT :rem 219
580 ROW=(XX-7)/2+1:COL=(YY-6)/2+1
  :rem 108
590 BX(RO,CO)=BX(RO,CO)+1:TY$="N"
  :rem 165
600 IFBX(RO,CO)=4THENBX$="X":GOSUB1480:GO
  SUB880:GOSUB950 :rem 223
610 IFDI$="A"THENBX(ROW-1,COL)=BX(ROW-1,C
  OL)+1:TY$="A":IFBX(RO-1,CO)=4GOTO650
  :rem 122
620 IFDI$="A"GOTO660 :rem 111
630 BX(ROW,COL-1)=BX(ROW,COL-1)+1:TY$="D"
  :rem 152
640 IFBX(ROW,COL-1)<4GOTO660 :rem 247
650 BX$="X":GOSUB1480:GOSUB880:GOSUB950
  :rem 107
660 D1=0:D2=0:IFBX$=""THENPL=PL+1:GOSUB95
  0 :rem 251
670 IFPC$(PL)="X"GOTO1210 :rem 166
680 POKEV,CH:POKEV+C,14:OV=V:GOTO150
  :rem 19
690 E=X:FORK=1TO9:E=E-2:IFE=0THENGOSUB144
  0:GOTO150 :rem 139
700 NEXT:GOTO240 :rem 223
710 PRINT"{CLR}":PRINTTAB(8)"(RVS) [<3>]BE
  NVENUTI A CONNECT'EM ":PRINT:PRINT:GO
  SUB1480 :rem 216
720 TS=84:GOSUB1460 :rem 110
730 PRINT"{BLU}PER FAR GIOCARE IL COMPUTE
  R, [<3>]C-64" :rem 228
740 PRINT"{BLU}PER 1 O 2 GIOCATORI, I LOR
  O NOMI...":PRINT:PRINT:FORT=1TO2500:N
  EXT :rem 135
750 FORI=1TO2:GOSUB1460:PRINT"{BLU}NOME D
  EL GIOC.N."I"{SIN} [<3>]";:INPUTPL$(I)
  :rem 173
755 IF PL$(I)="" THEN GOTO 750 :rem 15
760 PL$(I)=LEFT$(PL$(I),8):NEXT :rem 11
770 PRINT:GOSUB1460:F1=-1:F2=0:PRINT:FORT
  =1TO2:Z$=LEFT$(PL$(I),1):A=ASC(Z$)
  :rem 79

```

```

780 FL(T)=A-64:F1=F1+4:NC(T)=F1:IFPL$(T)=
"C-64"THENF2=F2+1:PC$(T)="X":rem 131
790 IFT=2ANDF2=2THENPL$(T)="COMMODORE"
:rem 102
800 PRINT"[<3>]{ 2 SPAZI}";PL$(T);"{BLU}
USA{ 2 SPAZI}IL{ 2 SPAZI}JOYSTICK";T:
NEXT :rem 51
810 L%=LEN(PL$(1)):M%=LEN(PL$(2)):L%=(L%/
2)+7:M%=(M%/2)+23 :rem 31
820 PRINT:PRINT:PRINT"{BLU}CAMPO REGOLARE
O MINI(R/M)?" :rem 179
830 GETZ$:IFZ$="R"GOTO860 :rem 234
840 IFZ$<>"M"GOTO830 :rem 136
850 LX=18:UX=7:TS=42 :rem 198
860 FORX=1TOUX:DT$=DT$+" ":NEXT:rem 160
870 GOSUB1480:GOSUB950:RETURN :rem 94
880 IFTY$="N"THENAA=RO:BB=CO :rem 235
890 IFTY$="A"THENAA=RO-1:BB=CO :rem 61
900 IFTY$="D"THENAA=RO:BB=CO-1 :rem 56
910 A=(AA*2+7)-2:B=BB*2+4:E=S+A*40+B+41
:rem 6
920 AA=FL(PL):CL=NC(PL) :rem 93
930 POKEE,AA+128:POKEE+C,CL:CL=2:RETURN
:rem 173
940 PRINT"{HOME}":FORJ=1TOCC:PRINT:NEXT:R
ETURN :rem 221
950 IFIN$=" "THENBX$=" ":PRINT"{CLR}"
:rem 117
960 PRINT"{HOME}":PRINTTAB(9)"{RVS}{BLU}
SCACCHIERA DI GIOCO{ 3 SPAZI}":PRINT
:rem 171
970 IFBX$<>"X"GOTO1000 :rem 253
980 IFPL=1THENP1=P1+1:GOTO1000 :rem 56
990 P2=P2+1:PL=2 :rem 132
1000 PRINTTAB(8)"{RVS}{CYN}"+PL$(1);TAB(2
4)"{RVS}{YEL}"+PL$(2):IFPL>2THENPL=1
:rem 73
1010 PRINTTAB(L%) "[<3>]";P1;TAB(M%)P2
:rem 213
1020 POKE781,6:SYS59903:NA$=PL$(PL):POKE6
46,NC(PL) :rem 193
1030 PRINT:PRINTTAB(4)NA$;" MUOVE ":IFPC$
(PL)=" "THENGOSUB1460 :rem 154
1040 IFIN$<>" "GOTO1070 :rem 206
1050 IN$="X":CC=6:GOSUB940 :rem 228
1060 FORX=1TO8:PRINTTAB(6)"{RED}";DT$:PRI
NT:NEXT:TI$="000000" :rem 167
1070 GOSUB1200 :rem 11
1080 A=P1+P2:IFA=TS:GOTO1100 :rem 94
1090 RETURN :rem 170
1100 WN$=" VINCE!":NA$=PL$(1)+WN$:IFP2=P1
THENNA$="GIOCO PARI!":GOTO1120 :rem 132
1110 POKE646,NC(1):IFP2>P1THENNA$=PL$(2)+
WN$:POKE646,NC(2) :rem 122
1120 CC=22:GOSUB940 :rem 66
1130 FORX=1TO7:POKE781,22:SYS59903:GOSUB1
480:PRINTTAB(12)"{SU}";NA$ :rem 196
1140 FORY=1TO300:NEXT:NEXT :rem 156
1150 PRINTTAB(7)"[<3>]UN'ALTRA PARTITA
{ 3 SPAZI}(S/N)?" :rem 64
1160 GETZ$:IFZ$=" "GOTO1160 :rem 239
1170 IFZ$="N"THENEND :rem 171
1180 IFZ$<>"S"THENGOTO1160 :rem 24
1190 FORA=1TO7:FORB=1TO12:BX(A,B)=0:NEXT:
NEXT:D1=0:D2=0:GOTO100 :rem 130
1200 PRINT"{HOME}":PRINT"{PUR}";LEFT$(TI$
,2);";":MID$(TI$,3,2);";":RIGHT$(TI$
,2):RETURN :rem 41
1210 NZ=0:VA=-1:NX=0:GOSUB1510 :rem 19
1220 VA=VA+1:NL=VA+1 :rem 144
1230 IFBZ$(VA)="X"GOTO1220 :rem 212
1240 FORA=1TO7:FORB=1TOUX-1 :rem 182
1250 IFBX(A,B)=3THENNZ=0:NL=4:NX=0:GOSUB1
380:A=7:B=UX-1:GOTO1270 :rem 173
1260 IFBX(A,B)=VATHENGOSUB1380 :rem 65
1270 NEXT:NEXT:NX=0:IFNZ>0GOTO1290
:rem 179
1280 BZ$(VA)="X":GOTO1220 :rem 132
1290 R=INT(RND(1)*NZ)+1:IFNX=NZTHENNZ=0:G
OTO1220 :rem 251
1300 IFZO(R)=99GOTO1290 :rem 83
1310 NX=NX+1:LOC=ZO(R) :rem 75
1320 FORZ=1TO4:RX$(Z)=" ":NEXT:RX=0
:rem 74
1330 IFRX=4THENZO(R)=99:GOTO1290:rem 218
1340 Z=INT(RND(1)*4)+1:IFRX$(Z)="X"GOTO13
40 :rem 215
1350 RX=RX+1:RX$(Z)="X" :rem 66
1360 E=PEEK(LOC+ZX(0,Z)):IFE=32THENGOSUB1
390:GOTO1420 :rem 234
1370 GOTO1330 :rem 203
1380 E=(A*2+7)-2:K=B*2+4:LOC=S+E*40+K:NZ=
NZ+1:ZO(NZ)=LOC:RETURN :rem 12
1390 XX=LOC-S:FORK=1TO21:XX=XX-40:YY=XX:I
FYY<40THENXX=K:K=21 :rem 224
1400 NEXT :rem 4
1410 ROW=(XX-7)/2+1:COL=(YY-6)/2+1:RETURN
:rem 175
1420 IFBX(ROW+ZX(Z,1),COL+ZX(Z,2))<NLTHEN
D1=LOC+ZX(Z,3):D2=LOC+ZX(Z,4):GOTO36
0 :rem 73
1430 GOTO1330 :rem 200
1440 GOSUB1570:POKEC+5,45:POKEC+6,165:POK
EC+4,33:POKEC+1,6:POKEC,5 :rem 160
1450 FORT=1TO200:NEXT:POKEC+4,32:POKEC+1,
0:POKEC,0:RETURN :rem 67
1460 GOSUB1570:POKEC+5,0:POKEC+6,247:POKE
C+4,17 :rem 245
1470 POKEC+1,40:POKEC,0:FORT=1TO100:NEXT:
POKEC+4,16:RETURN :rem 122
1480 GOSUB1570:POKEC+5,0:POKEC+6,255:POKE
C+4,21 :rem 241
1490 POKEC+15,40:FORA=1TO4:F1=20:FORB=1TO
2:F2=100:POKEC+1,F1:FORK=1TO2
:rem 216
1500 POKEC+15,F2:F1=F1*1.05:F2=F2*.9:NEXT
K:NEXTB:NEXTA:POKEC+6,15:RETURN :rem 170
1510 GOSUB1570:POKEC+5,6:FORZ=1TO15
:rem 116
1520 A=INT(10*RND(1))+1:Q=INT(25*RND(1))+
1:POKEC+1,40:F2=INT(15*RND(1))+1
:rem 59
1530 POKEC+15,F2:POKEC+4,23:FORT=1TOQ:NEX
T:POKEC+4,128:NEXTZ:RETURN :rem 27
1540 GOSUB1570:POKEC+5,8:POKEC+6,255:POKE
C+4,23 :rem 248
1550 POKEC+15,40:FORZ=1TO4:FORF1=255TO0ST
EP-6:POKEC+1,F1:NEXTF1 :rem 156
1560 FORF1=0TO255STEP10:POKEC+1,F1:NEXTF1
:POKEC+6,10:RETURN :rem 194
1570 FORL=CTOC+23:POKE1,0:NEXT:POKEC+24,1
5:RETURN :rem 105

```

Decipede

di J. Knesek
trad. e adatt. di S. Colombo

Ecco un gioco d'azione realizzato in BASIC e linguaggio macchina, che metterà alla prova le vostre doti di prontezza e riflessi. E' richie-

lizzo di un joystick in Porta 2.

In una splendida giornata primaverile decidete di fare un buon pic nic sui prati in cui sta spuntando la prima tenera erbetta e, appena trovato il posto che fa per voi, vi accingete a sistemare sul prato le provviste portate da casa. Improvvisamente, un inaspettato silenzio si crea tutt'attorno a voi: gli uccelli cessano di cinguettare, gli scoiattoli scompaiono improvvisamente, e una scura massa di nubi nere si avvicina sempre più velocemente.

Un giardiniere passa correndo vicino a voi, mentre continua a girarsi per scrutare dietro di sé, e vi urla: "Meglio che ve ne andiate!".

Mentre state cercando di capire il senso di quanto sta avvenendo attorno a voi, spunta un lungo millepiedi da dietro un albero e, in quattro e quattr'otto, divora il vostro pranzo cucinato con tanta cura.

La creatura, apparentemente ancora affamata, inizia a rivolgere la sua attenzione verso di voi: è stata una vera fortuna aver portato con sé la bombolletta di insetticida!

Scopo del gioco è infatti distruggere tutte le strane creature che vi stanno assalendo utilizzando l'insetticida, prima che esse possano privarvi di



quest'ultima entrando in contatto con lo spruzzatore.

In corrispondenza dell'inizio di ogni fase del gioco entrate in possesso di tre spruzzatori di insetticida per distruggere i millepiedi e le altre creature.

I vostri nemici sono tutt'altro che sem-

plici da sconfiggere: particolarmente pericolosa è infatti la cavalletta, poiché il suo movimento in diagonale la rende difficile da evitare e da distruggere. Le ondate di attacco hanno termine soltanto dopo l'intera distruzione del millepiedi (dieci segmenti), in corri-

spondenza della quale si ha una variazione dei colori delle creature che popolano lo schermo insieme ad un incremento della loro velocità.

Il punteggio che potete accumulare varia da un minimo di 25 punti ad un massimo di 1200, a seconda del tipo di creatura che riuscite a distruggere. Potete guadagnare un ulteriore spruzzatore di insetticida ogni 12000 punti raggiunti.

La vostra arma, lo spruzzatore, può essere spostato liberamente entro le quattro righe più basse dello schermo ed è dotato di un movimento estremamente rapido. Lo spruzzo dei proiettili

di insetticida avviene premendo il pulsante di fuoco sul joystick.

Note per il caricamento e la digitazione

Il programma è composto da una due parti a caricamento automatico.

Decipede viene caricato e mandato in esecuzione eseguendo un LOAD del primo programma e digitando quindi RUN.

Gli utilizzatori di disk drive dovranno sostituire, nella linea 130 del Programma 1, il valore 1 seguente l'istruzione LOAD con un 8.

Si noti che è possibile ottenere un

elenco completo di tutti i programmi presenti sulla cassetta e della loro posizione ponendo la cassetta completamente riavvolta nel registratore e digitando:

VERIFY *** (RETURN)

Poiché non esiste alcun programma dal nome ***, il computer visualizzerà sullo schermo i nomi di tutti i programmi incontrati (segnarsi la posizione del contanastro sottraendo due o tre unità, per ogni apparizione del messaggio FOUND...) fino al termine del nastro.

Decipede

Programma 1

1000 PRINT "{CLR}ATTENDERE..."	:rem	75	1250 DATA 000,000,000,000,160,000	:rem	37
1010 FORI=49152 TO 51487	:rem	224	1260 DATA 169,081,145,002,198,002	:rem	88
1020 READA:POKEI,A:NEXT	:rem	184	1270 DATA 166,002,224,255,208,002	:rem	79
1030 LOAD"DECIPEDE2",1	:rem	234	1280 DATA 198,003,177,002,201,032	:rem	75
1040 DATA 169,080,133,002,169,004	:rem	80	1290 DATA 240,033,201,088,208,014	:rem	80
1050 DATA 133,003,169,070,133,253	:rem	78	1300 DATA 024,165,002,105,041,133	:rem	64
1060 DATA 169,004,133,254,169,192	:rem	95	1310 DATA 002,144,002,230,003,076	:rem	61
1070 DATA 141,020,207,169,001,141	:rem	70	1320 DATA 115,192,032,104,193,234	:rem	79
1080 DATA 000,207,169,193,141,009	:rem	84	1330 DATA 160,000,165,156,208,003	:rem	73
1090 DATA 207,160,000,169,065,145	:rem	85	1340 DATA 076,156,192,132,156,032	:rem	90
1100 DATA 251,169,000,141,003,207	:rem	66	1350 DATA 230,192,162,130,142,019	:rem	78
1110 DATA 133,156,141,001,207,141	:rem	66	1360 DATA 207,076,014,196,000,000	:rem	75
1120 DATA 008,207,141,017,207,141	:rem	72	1370 DATA 000,000,000,000,169,128	:rem	60
1130 DATA 010,207,032,016,201,234	:rem	61	1380 DATA 141,018,212,169,009,141	:rem	85
1140 DATA 234,234,160,000,169,081	:rem	78	1390 DATA 019,212,169,002,141,015	:rem	80
1150 DATA 145,002,230,002,166,002	:rem	63	1400 DATA 212,169,129,141,018,212	:rem	80
1160 DATA 208,002,230,003,177,002	:rem	67	1410 DATA 096,000,000,000,000,000	:rem	43
1170 DATA 201,032,240,033,201,088	:rem	70	1420 DATA 000,000,108,008,207,000	:rem	55
1180 DATA 208,014,024,165,002,105	:rem	73	1430 DATA 000,000,000,000,000,000	:rem	30
1190 DATA 039,133,002,144,002,230	:rem	70	1440 DATA 000,198,003,096,000,000	:rem	67
1200 DATA 003,076,185,192,032,104	:rem	77	1450 DATA 000,000,000,000,000,000	:rem	32
1210 DATA 193,234,160,000,165,156	:rem	79	1460 DATA 000,000,000,000,160,000	:rem	40
1220 DATA 208,003,076,086,192,132	:rem	85	1470 DATA 177,253,201,081,240,019	:rem	87
1230 DATA 156,032,230,192,162,062	:rem	79	1480 DATA 024,165,253,105,039,133	:rem	88
1240 DATA 142,019,207,076,014,196	:rem	89			

1490 DATA 253,144,002,230,254,162	:rem	82	1810 DATA 142,159,192,162,176,142	:rem	96
1500 DATA 057,142,008,207,076,084	:rem	89	1820 DATA 015,193,142,072,193,142	:rem	88
1510 DATA 193,160,000,169,032,145	:rem	80	1830 DATA 093,192,142,163,192,162	:rem	96
1520 DATA 253,230,253,166,253,208	:rem	88	1840 DATA 198,142,017,193,142,074	:rem	99
1530 DATA 002,230,254,096,000,000	:rem	64	1850 DATA 193,142,095,192,142,165	:rem	101
1540 DATA 000,000,000,000,000,000	:rem	32	1860 DATA 192,162,039,142,069,193	:rem	105
1550 DATA 000,000,000,000,000,000	:rem	33	1870 DATA 142,160,192,162,041,142	:rem	85
1560 DATA 000,160,000,177,253,201	:rem	69	1880 DATA 012,193,142,090,192,162	:rem	92
1570 DATA 081,240,019,024,165,253	:rem	88	1890 DATA 001,142,001,207,096,000	:rem	73
1580 DATA 105,041,133,253,144,002	:rem	75	1900 DATA 000,201,062,208,079,172	:rem	79
1590 DATA 230,254,162,000,142,008	:rem	77	1910 DATA 001,207,192,001,208,072	:rem	75
1600 DATA 207,076,027,193,160,000	:rem	80	1920 DATA 162,024,142,008,193,142	:rem	84
1610 DATA 169,032,145,253,198,253	:rem	99	1930 DATA 065,193,142,086,192,142	:rem	99
1620 DATA 166,253,224,255,208,002	:rem	86	1940 DATA 156,192,162,105,142,011	:rem	84
1630 DATA 198,254,096,000,000,000	:rem	76	1950 DATA 193,142,068,193,142,089	:rem	108
1640 DATA 201,042,208,005,133,156	:rem	76	1960 DATA 192,142,159,192,162,144	:rem	102
1650 DATA 076,071,200,201,090,208	:rem	79	1970 DATA 142,015,193,142,072,193	:rem	94
1660 DATA 003,076,200,200,201,003	:rem	61	1980 DATA 142,093,192,142,163,192	:rem	100
1670 DATA 208,005,160,041,076,225	:rem	85	1990 DATA 162,230,142,017,193,142	:rem	90
1680 DATA 200,201,002,208,003,076	:rem	70	2000 DATA 074,193,142,095,192,142	:rem	88
1690 DATA 225,200,201,094,208,005	:rem	80	2010 DATA 165,192,162,041,142,069	:rem	85
1700 DATA 133,156,076,011,199,201	:rem	86	2020 DATA 193,142,160,192,162,039	:rem	86
1710 DATA 086,208,005,133,156,076	:rem	92	2030 DATA 142,012,193,142,090,192	:rem	78
1720 DATA 026,198,201,066,208,009	:rem	92	2040 DATA 096,000,000,000,141,014	:rem	54
1730 DATA 140,007,207,133,156,234	:rem	82	2050 DATA 207,142,015,207,140,016	:rem	72
1740 DATA 076,051,197,201,065,208	:rem	94	2060 DATA 207,172,000,207,208,003	:rem	71
1750 DATA 006,160,000,140,000,207	:rem	62	2070 DATA 076,225,194,173,000,220	:rem	82
1760 DATA 096,201,060,208,080,162	:rem	87	2080 DATA 160,000,162,000,142,013	:rem	59
1770 DATA 056,142,008,193,142,065	:rem	94	2090 DATA 207,074,176,001,136,074	:rem	89
1780 DATA 193,142,086,192,142,156	:rem	103	2100 DATA 176,001,200,074,176,001	:rem	68
1790 DATA 192,162,233,142,011,193	:rem	90	2110 DATA 202,074,176,001,232,074	:rem	74
1800 DATA 142,068,193,142,089,192	:rem	101	2120 DATA 176,003,238,013,207,142	:rem	77

2130 DATA 011,207,140,012,207,172	:rem	66	2450 DATA 066,240,004,140,000,207	:rem	69
2140 DATA 007,207,192,001,208,011	:rem	70	2460 DATA 096,238,000,207,169,065	:rem	98
2150 DATA 032,223,196,172,000,207	:rem	77	2470 DATA 145,251,096,000,000,000	:rem	68
2160 DATA 208,019,076,225,194,172	:rem	97	2480 DATA 000,172,000,207,192,001	:rem	68
2170 DATA 013,207,240,011,032,195	:rem	73	2490 DATA 208,001,096,160,000,169	:rem	86
2180 DATA 196,172,000,207,208,003	:rem	81	2500 DATA 032,145,251,024,165,251	:rem	78
2190 DATA 076,225,194,174,011,207	:rem	93	2510 DATA 105,040,133,251,144,002	:rem	66
2200 DATA 240,021,224,001,208,006	:rem	60	2520 DATA 230,252,160,000,177,251	:rem	75
2210 DATA 032,218,195,076,199,194	:rem	104	2530 DATA 201,032,240,012,201,088	:rem	68
2220 DATA 032,157,195,172,000,207	:rem	80	2540 DATA 208,003,076,051,195,140	:rem	85
2230 DATA 208,003,076,225,194,172	:rem	88	2550 DATA 000,207,096,234,206,000	:rem	75
2240 DATA 012,207,240,013,192,001	:rem	65	2560 DATA 207,169,065,145,251,096	:rem	104
2250 DATA 208,006,032,097,195,076	:rem	96	2570 DATA 000,000,000,000,000,000	:rem	36
2260 DATA 225,194,032,037,195,173	:rem	96	2580 DATA 000,160,000,169,032,145	:rem	75
2270 DATA 003,207,208,034,165,162	:rem	83	2590 DATA 251,198,251,164,251,192	:rem	103
2280 DATA 234,201,030,016,006,238	:rem	75	2600 DATA 255,208,003,198,252,234	:rem	91
2290 DATA 003,207,076,008,195,201	:rem	86	2610 DATA 160,000,177,251,201,032	:rem	69
2300 DATA 060,016,008,169,002,141	:rem	72	2620 DATA 240,019,201,088,208,003	:rem	80
2310 DATA 003,207,076,008,195,201	:rem	79	2630 DATA 076,224,195,201,062,208	:rem	90
2320 DATA 090,016,005,169,003,141	:rem	75	2640 DATA 003,076,224,195,140,000	:rem	78
2330 DATA 003,207,173,014,207,174	:rem	79	2650 DATA 207,096,234,169,065,145	:rem	105
2340 DATA 015,207,172,016,207,076	:rem	85	2660 DATA 251,096,000,000,000,000	:rem	59
2350 DATA 049,234,000,000,000,000	:rem	54	2670 DATA 000,000,000,000,000,000	:rem	37
2360 DATA 000,000,000,000,000,000	:rem	33	2680 DATA 000,000,160,000,169,032	:rem	66
2370 DATA 000,000,000,000,000,000	:rem	34	2690 DATA 145,251,230,251,164,251	:rem	89
2380 DATA 000,172,000,207,192,004	:rem	70	2700 DATA 192,000,208,003,230,252	:rem	70
2390 DATA 208,001,096,160,000,169	:rem	85	2710 DATA 234,160,000,177,251,201	:rem	74
2400 DATA 032,145,251,056,165,251	:rem	82	2720 DATA 032,240,019,201,088,208	:rem	83
2410 DATA 233,040,133,251,176,002	:rem	72	2730 DATA 003,076,163,195,201,060	:rem	84
2420 DATA 198,252,160,000,177,251	:rem	87	2740 DATA 208,003,076,163,195,140	:rem	91
2430 DATA 201,032,240,015,201,088	:rem	70	2750 DATA 000,207,096,234,169,065	:rem	96
2440 DATA 208,003,076,111,195,201	:rem	79	2760 DATA 145,251,096,000,000,000	:rem	70

2770 DATA 165,254,201,008,144,003	:rem	84	3090 DATA 004,212,169,009,141,005	:rem	79
2780 DATA 076,038,196,173,006,207	:rem	105	3100 DATA 212,169,170,141,001,212	:rem	67
2790 DATA 201,010,144,003,076,038	:rem	80	3110 DATA 169,129,141,004,212,056	:rem	81
2800 DATA 196,172,000,207,208,013	:rem	81	3120 DATA 165,004,233,040,133,004	:rem	67
2810 DATA 120,169,049,141,020,003	:rem	76	3130 DATA 176,002,198,005,160,000	:rem	75
2820 DATA 169,234,141,021,003,088	:rem	87	3140 DATA 177,004,201,081,208,003	:rem	74
2830 DATA 096,172,004,207,192,000	:rem	85	3150 DATA 076,051,197,201,088,208	:rem	96
2840 DATA 208,003,076,072,196,162	:rem	96	3160 DATA 003,076,095,197,201,094	:rem	95
2850 DATA 255,202,224,000,208,251	:rem	79	3170 DATA 208,003,076,118,199,201	:rem	91
2860 DATA 136,076,054,196,160,000	:rem	93	3180 DATA 087,208,003,076,122,197	:rem	97
2870 DATA 177,002,201,065,208,003	:rem	83	3190 DATA 201,086,208,003,076,118	:rem	88
2880 DATA 076,038,196,174,003,207	:rem	104	3200 DATA 199,201,042,208,003,076	:rem	81
2890 DATA 208,003,108,019,207,173	:rem	93	3210 DATA 160,200,160,040,177,004	:rem	67
2900 DATA 021,207,205,022,207,240	:rem	71	3220 DATA 201,066,208,004,169,032	:rem	79
2910 DATA 006,238,022,207,108,019	:rem	85	3230 DATA 145,004,160,000,169,066	:rem	79
2920 DATA 207,140,022,207,224,001	:rem	71	3240 DATA 145,004,160,001,140,007	:rem	65
2930 DATA 208,063,174,017,207,208	:rem	94	3250 DATA 207,096,000,000,000,234	:rem	65
2940 DATA 052,173,063,198,201,041	:rem	90	3260 DATA 032,230,192,160,040,177	:rem	81
2950 DATA 208,021,162,031,134,163	:rem	82	3270 DATA 004,201,066,208,004,169	:rem	83
2960 DATA 162,007,134,164,162,039	:rem	95	3280 DATA 032,145,004,160,000,238	:rem	74
2970 DATA 142,063,198,162,041,142	:rem	95	3290 DATA 006,207,173,006,207,201	:rem	80
2980 DATA 149,198,076,163,196,162	:rem	121	3300 DATA 010,048,003,140,000,207	:rem	58
2990 DATA 248,134,163,162,006,134	:rem	97	3310 DATA 140,007,207,076,202,192	:rem	79
3000 DATA 164,162,041,142,063,198	:rem	84	3320 DATA 000,000,000,000,000,000	:rem	30
3010 DATA 162,039,142,149,198,032	:rem	91	3330 DATA 000,160,040,177,004,201	:rem	64
3020 DATA 000,198,108,019,207,032	:rem	78	3340 DATA 066,208,004,169,032,145	:rem	89
3030 DATA 026,198,108,019,207,224	:rem	90	3350 DATA 004,160,000,169,032,145	:rem	75
3040 DATA 002,208,006,032,190,198	:rem	80	3360 DATA 004,238,005,207,140,007	:rem	77
3050 DATA 108,019,207,032,000,200	:rem	65	3370 DATA 207,076,202,192,160,040	:rem	84
3060 DATA 108,019,207,000,000,000	:rem	59	3380 DATA 177,004,201,066,208,004	:rem	84
3070 DATA 000,165,251,133,004,165	:rem	75	3390 DATA 169,032,145,004,160,000	:rem	79
3080 DATA 252,133,005,169,128,141	:rem	87	3400 DATA 140,007,207,096,169,080	:rem	89

3410 DATA 133,004,169,216,133,005		3730 DATA 081,208,010,224,002,240	
:rem	78	:rem	71
3420 DATA 173,002,207,160,000,145		3740 DATA 003,076,059,198,076,053	
:rem	70	:rem	105
3430 DATA 004,230,004,166,004,224		3750 DATA 198,201,066,208,003,076	
:rem	70	:rem	96
3440 DATA 000,208,003,230,005,234		3760 DATA 118,199,201,065,208,012	
:rem	65	:rem	94
3450 DATA 166,005,224,219,208,237		3770 DATA 140,000,207,140,017,207	
:rem	94	:rem	75
3460 DATA 166,004,224,232,208,231		3780 DATA 140,003,207,076,125,198	
:rem	83	:rem	96
3470 DATA 096,230,005,096,169,080		3790 DATA 201,088,208,007,140,003	
:rem	100	:rem	85
3480 DATA 133,004,169,004,133,005		3800 DATA 207,140,017,207,096,169	
:rem	80	:rem	95
3490 DATA 160,000,177,004,201,081		3810 DATA 086,145,163,076,174,199	
:rem	76	:rem	112
3500 DATA 240,042,201,066,240,038		3820 DATA 174,017,207,224,005,208	
:rem	74	:rem	87
3510 DATA 201,065,240,034,201,086		3830 DATA 006,206,018,207,076,059	
:rem	75	:rem	95
3520 DATA 240,030,201,094,240,026		3840 DATA 198,056,165,163,233,039	
:rem	71	:rem	108
3530 DATA 201,042,240,022,230,004		3850 DATA 133,163,176,002,198,164	
:rem	61	:rem	100
3540 DATA 166,004,224,000,208,003		3860 DATA 238,017,207,160,000,177	
:rem	72	:rem	91
3550 DATA 230,005,234,224,231,208		3870 DATA 163,201,081,208,003,076	
:rem	78	:rem	88
3560 DATA 215,166,005,224,007,208		3880 DATA 145,198,076,091,198,000	
:rem	87	:rem	110
3570 DATA 209,096,169,032,145,004		3890 DATA 000,000,000,000,000,000	
:rem	98	:rem	42
3580 DATA 076,218,197,230,005,096		3900 DATA 000,000,000,000,000,000	
:rem	104	:rem	34
3590 DATA 000,000,000,119,000,255		3910 DATA 000,000,000,000,174,017	
:rem	62	:rem	55
3600 DATA 162,000,142,018,207,162		3920 DATA 207,240,003,076,011,199	
:rem	74	:rem	88
3610 DATA 006,142,017,207,162,026		3930 DATA 162,050,142,024,207,165	
:rem	79	:rem	85
3620 DATA 142,024,207,234,234,234		3940 DATA 251,133,163,165,252,133	
:rem	82	:rem	91
3630 DATA 234,234,234,234,234,076		3950 DATA 164,173,000,207,141,017	
:rem	92	:rem	84
3640 DATA 059,198,160,000,177,163		3960 DATA 207,162,003,056,165,163	
:rem	99	:rem	94
3650 DATA 201,086,208,004,169,032		3970 DATA 233,240,133,163,176,002	
:rem	88	:rem	88
3660 DATA 145,163,174,018,207,240		3980 DATA 198,164,202,208,242,024	
:rem	93	:rem	99
3670 DATA 003,076,132,198,174,017		3990 DATA 173,017,207,105,018,141	
:rem	98	:rem	92
3680 DATA 207,224,001,208,006,238		4000 DATA 017,207,174,017,207,056	
:rem	86	:rem	83
3690 DATA 018,207,076,145,198,024		4010 DATA 165,163,233,040,133,163	
:rem	105	:rem	78
3700 DATA 165,163,105,041,133,163		4020 DATA 176,002,198,164,232,224	
:rem	82	:rem	88
3710 DATA 144,002,230,164,206,017		4030 DATA 024,208,240,142,017,207	
:rem	76	:rem	75
3720 DATA 207,160,000,177,163,201		4040 DATA 076,058,199,160,000,177	
:rem	78	:rem	97

4050 DATA 163,201,094,240,006,174	:rem	81	4370 DATA 025,142,024,207,238,024	:rem	84
4060 DATA 017,207,076,047,199,174	:rem	104	4380 DATA 207,096,169,016,141,011	:rem	92
4070 DATA 017,207,224,023,240,004	:rem	73	4390 DATA 212,169,012,141,012,212	:rem	76
4080 DATA 224,006,016,007,169,032	:rem	83	4400 DATA 169,015,141,013,212,174	:rem	79
4090 DATA 145,163,076,047,199,169	:rem	114	4410 DATA 024,207,142,008,212,160	:rem	73
4100 DATA 088,145,163,224,001,208	:rem	82	4420 DATA 017,140,011,212,206,024	:rem	66
4110 DATA 007,140,003,207,140,017	:rem	65	4430 DATA 207,096,000,000,000,000	:rem	57
4120 DATA 207,096,024,165,163,105	:rem	87	4440 DATA 000,000,000,000,000,000	:rem	34
4130 DATA 040,133,163,144,002,230	:rem	67	4450 DATA 000,050,174,017,207,240	:rem	75
4140 DATA 164,206,017,207,160,000	:rem	74	4460 DATA 003,076,029,200,169,035	:rem	89
4150 DATA 177,163,201,081,208,003	:rem	82	4470 DATA 141,024,207,169,144,133	:rem	90
4160 DATA 076,058,199,201,065,208	:rem	102	4480 DATA 163,169,005,133,164,169	:rem	103
4170 DATA 012,140,000,207,140,017	:rem	64	4490 DATA 001,141,017,207,076,078	:rem	91
4180 DATA 207,140,003,207,076,106	:rem	81	4500 DATA 200,160,000,177,163,201	:rem	68
4190 DATA 199,201,066,208,003,076	:rem	96	4510 DATA 042,208,027,160,001,177	:rem	80
4200 DATA 118,199,169,094,145,163	:rem	106	4520 DATA 163,201,088,240,013,201	:rem	75
4210 DATA 076,214,199,000,000,000	:rem	68	4530 DATA 090,240,009,160,000,169	:rem	81
4220 DATA 000,000,160,040,177,004	:rem	60	4540 DATA 032,145,163,076,064,200	:rem	85
4230 DATA 201,066,208,004,169,032	:rem	81	4550 DATA 160,000,169,090,145,163	:rem	88
4240 DATA 145,004,160,000,169,032	:rem	74	4560 DATA 174,017,207,224,039,208	:rem	96
4250 DATA 145,004,174,003,207,224	:rem	79	4570 DATA 007,140,017,207,140,003	:rem	75
4260 DATA 001,208,006,238,023,207	:rem	78	4580 DATA 207,096,230,163,238,017	:rem	99
4270 DATA 076,155,199,238,023,207	:rem	105	4590 DATA 207,160,000,177,163,201	:rem	84
4280 DATA 238,023,207,140,017,207	:rem	85	4600 DATA 081,208,003,076,071,200	:rem	77
4290 DATA 140,003,207,140,007,207	:rem	75	4610 DATA 201,066,208,003,076,160	:rem	81
4300 DATA 076,202,192,000,000,000	:rem	58	4620 DATA 200,169,042,145,163,169	:rem	94
4310 DATA 000,000,000,000,169,016	:rem	53	4630 DATA 016,141,011,212,169,012	:rem	74
4320 DATA 141,011,212,169,012,141	:rem	69	4640 DATA 141,012,212,169,015,141	:rem	78
4330 DATA 012,212,169,015,141,013	:rem	72	4650 DATA 013,212,174,024,207,142	:rem	80
4340 DATA 212,174,024,207,142,008	:rem	80	4660 DATA 008,212,160,017,140,011	:rem	73
4350 DATA 212,169,017,141,011,212	:rem	76	4670 DATA 212,224,035,208,005,162	:rem	84
4360 DATA 232,224,031,208,005,162	:rem	78	4680 DATA 010,076,142,200,162,035	:rem	80

```

4690 DATA 142,024,207,096,255,255      :rem 102
4700 DATA 255,255,255,255,255,255      :rem 105
4710 DATA 255,255,255,255,255,255      :rem 106
4720 DATA 160,040,177,004,201,066      :rem 80
4730 DATA 208,004,169,032,145,004      :rem 85
4740 DATA 160,000,169,032,145,004      :rem 79
4750 DATA 238,010,207,140,003,207      :rem 78
4760 DATA 140,017,207,140,007,207      :rem 82
4770 DATA 076,202,192,000,000,000      :rem 69
4780 DATA 000,000,000,000,162,001      :rem 51
4790 DATA 142,025,207,162,169,142      :rem 97
4800 DATA 076,192,142,146,192,162      :rem 98
4810 DATA 006,142,107,192,142,177      :rem 90
4820 DATA 192,238,147,192,096,174      :rem 112
4830 DATA 025,207,224,012,048,020      :rem 78
4840 DATA 162,177,142,076,192,142      :rem 101
4850 DATA 146,192,162,156,142,107      :rem 98
4860 DATA 192,142,177,192,206,147      :rem 106
4870 DATA 192,096,238,025,207,192      :rem 109
4880 DATA 000,208,002,160,039,177      :rem 88
4890 DATA 002,076,136,193,255,255      :rem 105
4900 DATA 255,255,255,255,120,169      :rem 102
4910 DATA 092,141,020,003,169,194      :rem 88
4920 DATA 141,021,003,088,096,000      :rem 80
4930 DATA 000,255                      :rem 66

```

Programma 2

```

10 IF PEEK(49152)=169 THEN30      :rem 218
20 PRINT"(CLR)CARICARE PRIMA IL PROGRAMMA 1!":END      :rem 23
30 PRINT CHR$(8):GOTO580      :rem 138
40 M=3:W=1:C=1:S=25:P=25      :rem 106
50 PRINT"(CLR){YEL}PROIETT.";M;" PUNTI";N      :rem 140
   ;"{ 9 SPAZI}{BLK}{ 10 SH W}";      :rem 206
60 PRINT"{ 30 SH W}{ 10 SH Q}";      :rem 75
70 PRINT"{ 39 SPAZI}{ 1 SH X}";      :rem 12
80 FORI=1TO21

```

```

90 PRINT"( 1 SH X){ 38 SPAZI}{ 1 SH X}";      :rem 37
100 NEXTI      :rem 25
110 FORI=1TO150      :rem 105
120 R=RND(1)*679+1      :rem 180
130 POKE1144+R,88      :rem 166
140 NEXTI:POKE1984,88:POKE2023,60:POKE186      :rem 138
   4,62      :rem 176
150 GOSUB190      :rem 191
160 SYS 50572:POKE251,211:POKE252,7      :rem 156
170 SYS 49152      :rem 104
180 GOTO240      :rem 180
190 SYS 49680:SYS 51432:POKE49299,2      :rem 52
200 POKE52994,C:POKE52996,S      :rem 4
210 POKE 52998,0:POKE52997,0      :rem 117
220 POKE52999,0:SS=2:POKE53013,SS:POKE530      :rem 38
   15,0:POKE6,0      :rem 122
230 FORXX=54272TO54295:POKEXX,0:NEXT:POKE      :rem 209
   54296,15:RETURN      :rem 27
240 REM      :rem 215
250 IFPEEK(52998)<10THENM=M-1:IFM=0THENNPR      :rem 73
   INT"(CLR)":GOTO550      :rem 90
260 X=PEEK(52997):Y=PEEK(52998):SP=PEEK(5      :rem 225
   3015):SI=PEEK(53002)      :rem 147
270 N=N+X*25+Y*50+SP*200+SI*1200:rem      :rem 65
280 IFN=>12000*WTHENM=M+1:W=W+1      :rem 183
290 PRINT"(HOME){YEL}"TAB(9);M;TAB(17);N      :rem 169
   :rem 32
300 PRINT"(SU){BLK}{ 30 SH W){ 10 SH Q}"      :rem 161
   :rem 129
310 SYS 50614      :rem 174
320 FORI=1TOX+Y+13      :rem 201
330 R=RND(1)*679+1      :rem 66
340 POKE1144+R,88      :rem 239
350 NEXTI      :rem 190
360 POKE1110+Z,32      :rem 84
370 Z=RND(1)*31+1      :rem 85
380 POKE1110+Z,88      :rem 87
390 IFPEEK(52998)<10THENP=P-1:GOTO490      :rem 88
400 C=C+1:IFC=15THENC=1      :rem 89
410 IFC/3<>INT(C/3) THEN480      :rem 249
420 ONC/3GOTO430,440,450,460,470:rem      :rem 219
430 S=P-3:P=P-1:GOTO500      :rem 224
440 S=P-3:P=P-1:GOTO500      :rem 9
450 S=P-4:P=P-1:GOTO500      :rem 177
460 S=P-4:P=P-1:GOTO500      :rem 254
470 S=P-4:P=P-1:GOTO500      :rem 104
480 IFC<8THENS=P:GOTO500      :rem 104
490 S=P-1      :rem 205
500 IFS<7THENS=7      :rem 198
510 IFP=7THENP=13      :rem 36
520 GOSUB190
530 SYS 50572:POKE251,211:POKE252,7:SYS49      :rem 198
   152      :rem 36
540 GOTO240
550 PRINT"(CLR){ 15 DES){ 10 GIU'}(RED)GA      :rem 198
   ME OVER"      :rem 36
560 PRINT:PRINT:PRINT"{ 15 DES}PUNTI";N
570 FORS=1TO4000:NEXT

```

```

580 POKE53280,0:POKE53281,0:PRINT"(CLR)
{BLU}{ 7 GIU'"; :rem 97
590 PRINT"{ 4 SPAZI}{RVS}{ 2 SPAZI}{<*>}
{OFF}{RVS}{ 1 SH £}{<*>}{OFF}{RVS}
{RVS}{ 1 SH £}{<*>}{OFF}{RVS}
{ 1 SH £}{<*>}{OFF}{RVS}
{ 1 SH £}{<*>}{OFF}{RVS}
{ 1 SH £}{<*>}{OFF}{RVS}
{ 2 SPAZI}{<*>}{OFF}{RVS}{ 1 SH £}
{<*>}{OFF}" :rem 125
600 PRINT"{ 4 SPAZI}{RVS}{OFF}{RVS}
{OFF}{RVS}{OFF}{ 3 SPAZI}{RVS}
{OFF}{ 4 SPAZI}{RVS}{OFF}{ 2 SPAZI}
{RVS}{OFF}{RVS}{OFF}{RVS}{OFF}
{ 3 SPAZI}{RVS}{OFF}{RVS}{OFF}
{RVS}{OFF}" :rem 115
610 PRINT"{ 4 SPAZI}{RVS}{OFF}{RVS}
{OFF}{RVS}{ 2 SPAZI}{OFF}{ 2 SPAZI}
{RVS}{OFF}{ 4 SPAZI}{RVS}{OFF}
{ 2 SPAZI}{RVS}{ 2 SPAZI}{OFF}
{ 1 SH £}{RVS}{ 2 SPAZI}{OFF}
{ 2 SPAZI}{RVS}{OFF}{RVS}{OFF}
{RVS}{ 2 SPAZI}{OFF}" :rem 121
620 PRINT"{ 4 SPAZI}{RVS}{OFF}{RVS}
{OFF}{RVS}{OFF}{ 3 SPAZI}{RVS}
{OFF}{ 4 SPAZI}{RVS}{OFF}{ 2 SPAZI}
{RVS}{OFF}{ 3 SPAZI}{RVS}{OFF}
{ 3 SPAZI}{RVS}{OFF}{RVS}{OFF}
{RVS}{OFF}" :rem 209
630 PRINT"{ 4 SPAZI}{RVS}{ 2 SPAZI}{OFF}
{ 1 SH £}{<*>}{RVS}{OFF}
{ 1 SH £}{<*>}{RVS}{OFF}
{ 1 SH £}{<*>}{RVS}{OFF}
{ 1 SH £}{RVS}{OFF}{ 3 SPAZI}
{<*>}{RVS}{OFF}{ 1 SH £}{RVS}
{ 2 SPAZI}{OFF}{ 1 SH £}{<*>}{RVS}
{OFF}{ 1 SH £}{ 3 GIU'"; :rem 183
640 PRINT:PRINTSPC(8)"{RVS}FUOCO PER INIZ
IARE... {OFF}" :rem 141
650 IFPEEK(56320)=111THENRUN 40 :rem 248
660 GOTO650 :rem 112

```

SUPERCOMMODORE

La rubrica INPUT/OUTPUT è gratuita ed aperta a tutti i lettori. Chi desidera comprare, vendere o cambiare hardware o software

INPUT/OUTPUT

☐ COMPRO ☐ VENDO ☐ CAMBIO
☐ VIC 20 ☐ C 16 ☐ C64 ☐ PERIF. ☐ SOFTWARE

Nome Cognome

Via C.A.P.

Città Tel.



BIT regala Le Grandi Guide di BIT

Da gennaio e fino a dicembre '87 ogni mese Bit vi dà un grande appuntamento in edicola con le Grandi Guide di Bit: veri manuali di riferimento di eccezionale utilità, dedicati al personal computer, alla programmazione, ai sistemi operativi, ai linguaggi. A fine anno possederete una biblioteca ricca, aggiornata e completa: per lavorare meglio e conoscere più a fondo il vostro personal computer.



Ogni mese
Bit ti aspetta in edicola
con la sua Guida.



GRUPPO EDITORIALE
JACKSON
DIVISIONE PERIODICI

in EDICOLA!

IL GRANDE DIZIONARIO JACKSON DI ELETTRONICA E INFORMATICA

Il Grande
Dizionario
Jackson

di Elettronica è
informatica è il risultato di
un importante lavoro di
ricerca, durata più di
cinque anni, che ha
coinvolto decine di
specialisti.

DEI è l'unica opera al
mondo che risponde
all'ormai improrogabile
necessità di raccogliere e
organizzare tutti i termini
tecnici nati dalla diffusione
massiccia dell'elettronica,
informatica e
comunicazione per
formalizzare e consolidare
il "nuovo sapere".

DEI è per tutti coloro che
hanno capito l'importanza
di sapersi orientare in
questo nuovo universo
linguistico; per chi vuole

 Philips
Computers

comprendere il significato
di termini che le nuove
discipline hanno introdotto
anche nel linguaggio
comune; per quanti
desiderano saperli utilizzare
con padronanza per non
rimanere esclusi da un
mondo che sta compiendo

un cambiamento le cui
dimensioni sono senza
precedenti; per coloro,
infine, che per lavoro,
studio, informazione
vogliono possedere, già da
oggi, la "cultura di
domani".

Con DEI, infatti, la
trasformazione della
scienza di oggi nella
cultura di domani compie
un decisivo passo avanti.

- 45000 termini
 - 3000 illustrazioni
 - 1000 monografie
- in fascicoli settimanali
da rilegare in 12
splendidi volumi.



GRUPPO EDITORIALE
JACKSON
DIVISIONE GRANDI OPERE

IN COLLABORAZIONE CON

PHILIPS



WORK

Data Aid

di H. Pesch
trad. e adatt. di S. Colombo

Questa utility risulta particolarmente d'aiuto quando si lavora con programmi contenenti molte istruzioni DATA: aggiunge infatti al BASIC molti nuovi comandi appositamente studiati per la programmazione o l'edizione DATA.

La memorizzazione in linee DATA di numeri costituenti istruzioni in linguaggio macchina o dati per SPRITE e caratteri ridefiniti, consente di inserire in memoria questi valori nel momento più opportuno e durante l'esecuzione del programma, tramite istruzioni POKE.

Questa interessante routine permette di poter gestire in modo efficace ed efficiente tutti i programmi contenenti linee DATA: esaminiamo i nuovi comandi che DATA Aid aggiunge al BASIC del vostro C64:

!R (Restore): resetta i puntatori ai dati su una particolare linea di programma. La successiva istruzione READ inizierà a leggere i dati dal primo numero contenuto nella linea indicata. Questo comando è del tutto analogo all'istruzione "RESTORE linea" del BASIC 7.0. Ecco un esempio dell'uso del comando !R:

```
10 DATA GIOVANNI,MARIO,MAURO
20 DATA MARIA,LAURA,MARINA
30 DATA 20,23,19
40 !R 20:READ A$;!R 10:READ B$:
50 L=30:!R L: READ N
60 PRINT A$,B$,N
RUN
MARIA GIOVANNI 20
READY.
```

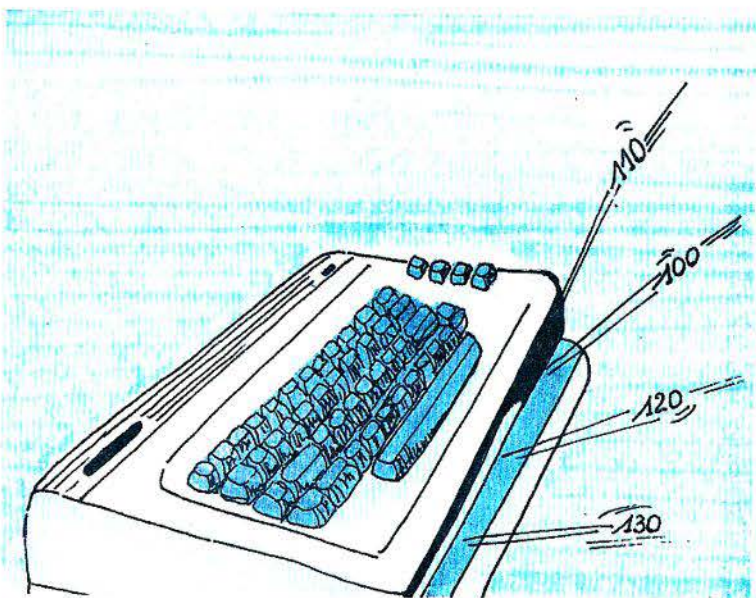
Come potete vedere, !R accetta il numero di linea come qualunque espressione numerica, anche una variabile. !R può inoltre essere utilizzato anche in modo diretto.

!S (Super POKE): trasferisce in memoria i numeri nelle istruzioni DATA, in maniera estremamente rapida. !S 100,5000 legge i byte contenuti nelle DATA (numeri interi in formato decima-

le, da 0 a 255) iniziando con la linea 100 e li inserisce in memoria a partire dalla locazione 5000.

Il procedimento ha termine quando viene letto un numero negativo. Questo metodo è più veloce di quello tradizionale (READ/POKE) di almeno quattro volte.

!S può essere usato all'interno di un programma, oppure in modo diretto; si



ricorda che può essere utilizzato soltanto quando i numeri nelle DATA siano interi e compresi tra 0 e 255, ed un numero negativo qualsiasi deve chiuderne la sequenza. Naturalmente quest'ultimo numero minore di zero non viene inserito in memoria.

ID (DATA automatiche): crea automaticamente delle linee DATA leggendo i dalla memoria i valori numerici. Questo comando è utile quando si è già caricato in memoria un programma in linguaggio macchina (oppure dati di SPRITE o caratteri) e si desidera trasformare i valori numerici (istruzioni l/m o byte di forma) in una sequenza di istruzioni DATA da inserire in un programma BASIC. Ad esempio:

ID 64,1000,49151

scrive 64 linee (di 16 numeri ciascuna)

di istruzioni DATA. Il primo numero di linea generato sarà 1001, e la prima locazione di memoria letta sarà la 49152. Il formato di questo comando è !D n,l,m dove n è il numero di linee DATA che devono essere scritte, iniziando dal numero di linea l-1 e contenenti i valori numerici presenti in memoria a partire dalla locazione m+1.

IM (Mega DATA): permette di avere una visione ingrandita di ciascun numero contenuto nelle linee DATA, per facilitarne la correzione.

IM 150 ad esempio visualizza il primo numero contenuto nella linea DATA 150, nella parte centrale dello schermo ed ingrandito quattro volte rispetto alle sue dimensioni normali. La pressione di RETURN farà visualizzare il numero successivo, e così via. Questo comando è estremamente utile per ricercare l'errore quando si ottengono degli "IL-

LEGAL QUANTITY ERROR" utilizzando letture di linee DATA.

IC (Cambia colore): permette di cambiare a piacere i colori dello schermo per rendere il lavoro più confortevole. IC 2,7,0 cambia ad esempio il colore del bordo in rosso, quello del fondo in giallo e quello del testo in nero.

Il formato di questo comando è IC bordo, fondo, testo, ed i codici colore devono naturalmente essere compresi tra 0 e 15. Funziona sia in modo diretto che da programma.

IB (Beep): emette un segnale acustico. IB G5 suona una nota G (nella dizione americana) nell'ottava 5. Si noti che questo comando richiede l'inserimento delle note secondo lo standard americano (dove le note sono C, D, F, G, e A) e le ottave variano da 0 a 7.

IZ (Zero): disabilita DATA Aid.

DATA Aid

```
100 PRINT "{CLR}ATTENDERE..." :rem 27
110 FORI=49152 TO 50183 :rem 168
120 READA:POKEI,A:NEXT :rem 136
130 PRINT "{CLR}" :rem 248
140 SYS49152 :rem 153
150 DATA 169,022,141,008,003,169,192 :rem 233
160 DATA 032,104,194,169,015,141,024 :rem 227
170 DATA 212,169,224,141,006,212,096 :rem 231
180 DATA 000,032,115,000,201,033,240 :rem 198
190 DATA 003,076,231,167,032,038,192 :rem 236
200 DATA 076,228,167,032,115,000,201 :rem 218
210 DATA 067,208,003,076,100,192,201 :rem 220
220 DATA 082,208,003,076,162,192,201 :rem 226
230 DATA 066,208,003,076,157,194,201 :rem 235
240 DATA 068,208,003,076,000,193,201 :rem 224
250 DATA 083,208,003,076,000,194,201 :rem 223
260 DATA 077,208,003,076,000,195,201 :rem 228
270 DATA 090,240,001,096,169,228,162 :rem 239
280 DATA 167,141,008,003,142,009,003 :rem 222
290 DATA 096,000,032,115,000,032,138 :rem 217
300 DATA 173,032,247,183,166,020,142 :rem 228
310 DATA 032,208,032,253,174,032,138 :rem 225
320 DATA 173,032,247,183,166,020,142 :rem 230
```

```
330 DATA 033,208,032,253,174,032,138 :rem 228
340 DATA 173,032,247,183,166,020,189 :rem 243
350 DATA 146,192,032,210,255,096,144 :rem 237
360 DATA 005,028,159,156,030,031,158 :rem 234
370 DATA 129,149,150,151,152,153,154 :rem 238
380 DATA 155,032,115,000,032,138,173 :rem 224
390 DATA 032,247,183,165,020,141,195 :rem 239
400 DATA 192,165,021,141,202,192,169 :rem 231
410 DATA 000,141,233,192,169,008,141 :rem 223
420 DATA 234,192,032,223,192,224,101 :rem 223
430 DATA 208,249,032,223,192,224,000 :rem 226
440 DATA 208,242,032,223,192,224,131 :rem 225
450 DATA 208,235,173,233,192,133,065 :rem 240
460 DATA 173,234,192,133,066,096,238 :rem 251
470 DATA 233,192,208,004,024,238,234 :rem 235
480 DATA 192,174,254,010,096,000,080 :rem 233
490 DATA 079,075,069,050,053,051,044 :rem 245
500 DATA 191,135,104,000,000,000,144 :rem 201
510 DATA 064,253,000,000,032,115,000 :rem 200
520 DATA 032,138,173,032,247,183,166 :rem 240
530 DATA 020,134,253,032,253,174,032 :rem 222
```

540 DATA 138,173,162,246,160,192,032	:rem 239	860 DATA 169,000,141,004,212,096,000	:rem 222
550 DATA 212,187,032,253,174,032,138	:rem 237	870 DATA 000,000,000,000,000,000,000	:rem 177
560 DATA 173,162,251,160,192,032,212	:rem 230	880 DATA 000,032,162,192,032,115,000	:rem 216
570 DATA 187,169,185,141,002,003,169	:rem 247	890 DATA 032,138,173,032,247,183,166	:rem 250
580 DATA 193,141,003,003,144,005,129	:rem 226	900 DATA 122,164,123,134,067,132,068	:rem 234
590 DATA 000,000,000,000,169,246,160	:rem 211	910 DATA 165,065,133,122,165,066,133	:rem 238
600 DATA 192,032,162,187,169,051,160	:rem 239	920 DATA 123,032,121,000,201,131,240	:rem 202
610 DATA 193,032,103,184,162,246,160	:rem 232	930 DATA 007,032,115,000,201,131,208	:rem 211
620 DATA 192,032,212,187,032,221,189	:rem 236	940 DATA 249,032,115,000,201,044,240	:rem 219
630 DATA 162,000,189,000,001,240,006	:rem 211	950 DATA 249,201,045,240,041,032,158	:rem 233
640 DATA 157,255,001,232,208,245,169	:rem 242	960 DATA 183,138,160,000,145,020,230	:rem 225
650 DATA 181,133,025,169,193,133,026	:rem 241	970 DATA 020,208,002,230,021,032,121	:rem 209
660 DATA 160,000,177,025,157,255,001	:rem 229	980 DATA 000,201,000,208,224,160,001	:rem 208
670 DATA 232,200,192,004,208,245,169	:rem 237	990 DATA 177,122,201,000,240,012,032	:rem 217
680 DATA 016,133,252,134,002,169,251	:rem 233	1000 DATA 115,000,032,115,000,032,115	:rem 242
690 DATA 160,192,032,162,187,169,051	:rem 248	1010 DATA 000,024,144,197,166,067,164	:rem 25
700 DATA 160,193,032,103,184,162,251	:rem 228	1020 DATA 068,134,122,132,123,096,000	:rem 11
710 DATA 160,192,032,212,187,032,013	:rem 224	1030 DATA 141,009,003,234,162,024,189	:rem 18
720 DATA 184,032,221,189,166,002,160	:rem 234	1040 DATA 130,194,032,210,255,202,208	:rem 11
730 DATA 001,185,000,001,240,007,157	:rem 214	1050 DATA 247,169,006,141,033,208,169	:rem 33
740 DATA 255,001,232,200,208,244,169	:rem 231	1060 DATA 006,141,032,208,096,032,068	:rem 22
750 DATA 044,157,255,001,232,198,252	:rem 242	1070 DATA 069,084,065,086,073,084,067	:rem 49
760 DATA 208,201,202,202,032,202,170	:rem 213	1080 DATA 065,032,068,073,065,045,065	:rem 34
770 DATA 032,134,164,068,065,084,065	:rem 248	1090 DATA 084,065,068,032,032,032,159	:rem 31
780 DATA 198,253,240,003,076,056,193	:rem 251	1100 DATA 147,008,000,000,032,115,000	:rem 244
790 DATA 169,131,141,002,003,169,164	:rem 237	1110 DATA 024,233,064,170,189,201,194	:rem 24
800 DATA 141,003,003,076,116,164,243	:rem 223	1120 DATA 133,252,169,001,133,253,032	:rem 13
810 DATA 127,227,008,247,000,247,008	:rem 234	1130 DATA 115,000,024,233,047,170,024	:rem 5
820 DATA 083,255,247,165,252,141,000	:rem 235	1140 DATA 006,253,038,252,144,002,230	:rem 12
830 DATA 212,165,253,141,001,212,169	:rem 228	1150 DATA 253,202,208,244,076,216,193	:rem 30
840 DATA 017,141,004,212,160,000,162	:rem 213	1160 DATA 000,000,000,000,000,000,195	:rem 233
850 DATA 000,232,208,253,200,208,248	:rem 228	1170 DATA 250,012,045,081,102,145,032	:rem 9

1180 DATA 068,229,032,115,000,201,000	:rem	6	1500 DATA 248,007,169,001,141,021,208	:rem	17
1190 DATA 208,003,076,116,164,032,115	:rem	22	1510 DATA 141,039,208,141,023,208,141	:rem	16
1200 DATA 000,032,115,000,170,032,115	:rem	245	1520 DATA 029,208,169,128,141,000,208	:rem	26
1210 DATA 000,032,205,189,096,000,201	:rem	6	1530 DATA 141,001,208,169,005,032,210	:rem	9
1220 DATA 064,048,002,233,064,096,170	:rem	24	1540 DATA 255,234,141,033,208,169,000	:rem	23
1230 DATA 189,253,194,141,139,195,181	:rem	46	1550 DATA 160,069,153,057,003,136,208	:rem	31
1240 DATA 252,096,064,065,066,032,162	:rem	32	1560 DATA 250,141,028,208,096,032,207	:rem	28
1250 DATA 192,234,169,147,032,210,255	:rem	31	1570 DATA 255,201,013,208,003,076,028	:rem	22
1260 DATA 166,122,164,123,134,067,132	:rem	25	1580 DATA 195,201,095,208,014,169,000	:rem	31
1270 DATA 068,165,065,133,122,165,066	:rem	37	1590 DATA 141,021,208,032,024,229,032	:rem	17
1280 DATA 133,123,032,174,195,160,032	:rem	22	1600 DATA 094,194,076,116,164,024,144	:rem	35
1290 DATA 132,252,132,253,132,254,032	:rem	19	1610 DATA 225,215,255,140,000,207,096	:rem	20
1300 DATA 115,000,201,000,240,023,201	:rem	238	1620 DATA 234,169,065	:rem	15
1310 DATA 044,208,003,076,077,195,032	:rem	27			
1320 DATA 236,194,166,253,134,252,166	:rem	38			
1330 DATA 254,134,253,133,254,024,144	:rem	23			
1340 DATA 226,032,208,194,032,115,000	:rem	13			
1350 DATA 201,131,208,249,234,169,255	:rem	33			
1360 DATA 072,169,002,072,169,001,072	:rem	26			
1370 DATA 169,000,072,104,201,255,208	:rem	20			
1380 DATA 003,076,224,195,032,243,194	:rem	33			
1390 DATA 168,162,255,142,136,195,162	:rem	44			
1400 DATA 207,142,137,195,024,173,136	:rem	28			
1410 DATA 195,105,008,144,003,238,137	:rem	25			
1420 DATA 195,141,136,195,136,208,239	:rem	41			
1430 DATA 162,008,160,036,032,154,195	:rem	25			
1440 DATA 189,151,209,153,066,003,136	:rem	33			
1450 DATA 136,136,202,208,244,032,164	:rem	24			
1460 DATA 195,024,144,192,000,169,000	:rem	23			
1470 DATA 141,014,220,169,051,133,001	:rem	11			
1480 DATA 096,169,055,133,001,169,001	:rem	33			
1490 DATA 141,014,220,096,169,013,141	:rem	24			

WORK

Come modificare
le icone del Geos

di D. Curtis
trad. e adatt. di R. Polini

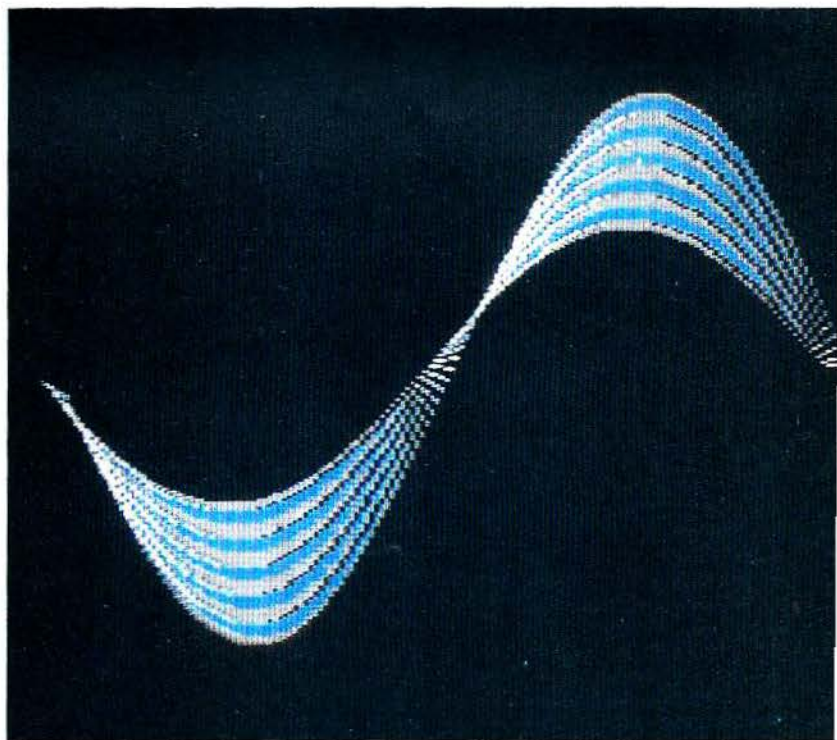
Il nuovo sistema operativo GEOS
Il C64 fa un largo uso di icone (o immagini grafiche) per l'individuazione dei programmi da utilizzare. Questo articolo mostra come procedere per modificarne l'aspetto, offrendone anche gli strumenti: un programma che permette di sostituire le icone standard con altre di nuovo disegno comprese nel programma stesso, oppure disegnandole personalmente. Naturalmente proposto in questo articolo è utilizzabile soltanto in ambiente GEOS con disk drive, e non con il normale sistema operativo del C64.

Il programma proposto è scritto interamente in BASIC: dopo averlo caricato, inserite nel drive il disco "GEOS work disk" (e non il "GEOS system disk"), quindi date il RUN.

Il programma vi chiederà di inserire il nome del file relativo all'icona che desiderate modificare, quindi leggerà da disco l'icona stessa e la visualizzerà sullo schermo.

Sottostante all'icona corrente, il programma visualizzerà la nuova icona proposta (i cui dati di forma sono contenuti nel programma stesso in istruzioni DATA). Potrete ora scegliere se effettuare o meno delle modifiche sulla nuova icona, rispondendo rispettivamente con S o con N alla domanda posta dal computer.

Infine, vi verrà richiesto se desiderate o meno assegnare la nuova forma dell'icona anche ad altre icone prees-



stenti: rispondete con S o con N a seconda dei vostri desideri.

Le nuove icone proposte automaticamente dal programma sono immagazzinate nelle linee DATA a partire dalla linea 1000. Dal momento che le icone utilizzate dal GEOS sono nel formato standard degli SPRITE, potete inseri-

re nel programma delle icone disegnate da voi stessi - sostituendo quelle attualmente esistenti - utilizzando uno SPRITE editor come "SPRITE Magic" apparso su SuperCommodore.

Il programma comprende, nella serie di istruzioni DATA, i dati relativi a cinque nuove icone: dal momento che

viene utilizzato un solo gruppo di DATA alla volta (una sola nuova icona) ed i gruppi sono tra loro separati da un'istruzione REM, l'utilizzo di una nuova icona diversa dalla prima richiede la cancellazione dal programma dei

gruppi di linee DATA precedenti a quello che si desidera utilizzare. Una nota importante: non cercate di modificare l'icona di un file standard del GEOS (ovvero uno di quelli indicati

nel menu "desktop" del GEOS), dal momento che i dati relativi a queste icone sono immagazzinati direttamente all'interno del sistema operativo GEOS

Modificatore di icone per GEOS

```

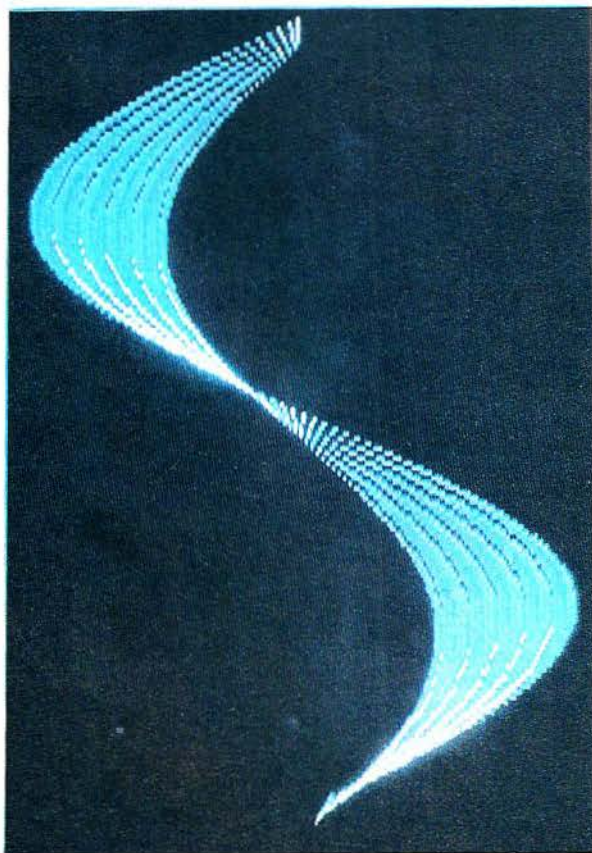
10 POKE53281,0:POKE53280,0:POKE53269,0:DN
   $="":IU=0 :rem 58
20 PRINT"{CLR}{RED}{GIU'}"SPC(10)"(RVS)ED
   ITOR DI ICONA" :rem 163
30 PRINT"{GIU'}"<4>"INSERIRE IL NOME DELL
   'ICONA SU DISCO DA MODIFICARE{GIU'}"
   :rem 54
40 INPUT IN$ :rem 175
50 PRINT"{GIU'}L'ICONA SCELTA E'
   { 4 SPAZI}>{RED}"IN$"<4>"<:PRINT"
   {GIU'}E' CORRETTO?{ 3 SPAZI}";
   :rem 236
60 INPUT" S/N";AN$:IF AN$<>"S"THEN10
   :rem 21
70 FORCT=1TO16:PD$=PD$+CHR$(32):NEXT:IN$=
   LEFT$(IN$+PD$,16) :rem 162
80 OPEN15,8,15,"I0" :rem 223
90 INPUT#15,ER$,EM$:IFER$<>"00"THENPRINTE
   R$+" "+EM$:CLOSE15:END :rem 219
100 T=18:S=1 :rem 134
110 OPEN2,8,2,"#" :rem 75
120 PRINT#15,"U1";2;0;T;S:FC=0:FP=2:BP=5
   :rem 65
130 PRINT#15,"B-P";2;0:GET#2,NT$,NS$:T=AS
   C(NT$+CHR$(0)):S=ASC(NS$+CHR$(0))
   :rem 205
140 PRINT#15,"B-P";2;FP:GET#2,FT$:FT$=FT$
   +CHR$(0):IFASC(FT$)=0THEN280:rem 246
150 PRINT#15,"B-P";2;BP :rem 21
160 FORCT=1TO16 :rem 141
170 GET#2,BI$ :rem 168
180 IFBI$=""THENBI$=CHR$(0) :rem 245
190 FI=ASC(BI$) :rem 61
200 IFFI>127THENFI=FI-128 :rem 218
210 IFFI<32THENFI=63 :rem 182
220 IFFI>96ANDFI<123THENFI=FIAND223
   :rem 135
230 IFFI=34THENFI=63 :rem 187
240 DN$=DN$+CHR$(FI) :rem 75
250 NEXTCT :rem 109
260 IFDN$=IN$THEN310 :rem 152
270 DN$="" :rem 208
280 FC=FC+1:IFFC=8ANDT<>0THEN120:rem 159
290 IFFC=8ANDT=0THEN560 :rem 134
300 FP=FP+32:BP=BP+32:GOTO140 :rem 191
310 GET#2,IT$,IS$ :rem 162
320 IT=ASC(IT$+CHR$(0)):IS=ASC(IS$+CHR$(0))
   :rem 173
330 PRINT#15,"U1";2;0;IT;IS :rem 41
340 PRINT#15,"B-P";2;5 :rem 185
350 FORCT=1TO63:GET#2,ID$:ID$=ASC(ID$+CHR$(0))
   :POKE831+CT,ID$:NEXT :rem 95
360 POKE53269,1:POKE2040,13:POKE53287,11:
   POKE53248,110:POKE53249,147 :rem 82
370 POKE53249,147 :rem 152
380 PRINTSPC(15)"{ 2 GIU'}{BLU}<-ICONA AT
   TUALE" :rem 45
390 RESTORE:FORCT=1TO63 :rem 242
400 READNI:POKE895+CT,NI :rem 219
410 NI$=NI$+CHR$(NI) :rem 92
420 NEXTCT :rem 108
430 POKE53269,3:POKE2041,14:POKE53288,11:
   POKE53250,110:POKE53251,180 :rem 68
440 PRINTSPC(15)"{ 3 GIU'}{RED}<-NUOVA IC
   ONA" :rem 177
450 IFIU=0THENPRINT"{ 2 GIU'}"<4>"OK PER
   CAMBIARLA{ 3 SPAZI}"; :rem 36
460 IFIU=0THENINPUT" S/N";AN$:IF AN$<>"S"
   THEN GOTO550 :rem 132
470 IFIU=0THENPRINT#15,"B-P";2;5:rem 134
480 IFIU=0THENPRINT#2,NI$; :rem 105
490 IFIU=0THENPRINT#15,"U2";2;0;IT;IS
   :rem 250
500 IFIU=0THENIU=1:PRINT"{ 11 SU}":GOTO33
   0 :rem 185
510 PRINT"{ 2 GIU'}"<4>"MODIFICA COMPLETA
   TA{ 10 SPAZI}":CLOSE2:CLOSE15
   :rem 62
520 PRINT"{GIU'}MODIFICA ANALOGA DI UN'AL
   TRA ICONA{ 18 SPAZI}"; :rem 24
530 INPUT" S/N";A$:IFA$="S"THEN10
   :rem 110
540 CLOSE2:CLOSE15:POKE53269,0:END
   :rem 54
550 CLOSE2:CLOSE15:POKE53269,0:PRINT"
   {CLR}":PRINTSPC(13)"TERMINATO!
   { 4 SPAZI}":END :rem 79
560 PRINT"{GIU'}FILE NOT FOUND":CLOSE2:CL
   OSE15:END :rem 19
570 REM ** DATI ICONA ** :rem 184
580 REM ** GEOPAINT ** :rem 128
590 DATA 255,255,255,146,73,1,255,255,125
   ,168,0,1,248,31,141,168 :rem 93
600 DATA 32,81,248,64,33,168,64,225,248,1
   31,17,171,255,253,248,136 :rem 196
610 DATA 17,168,112,33,248,64,33,169,160,
   65,250,31,129,136,0,1 :rem 232
620 DATA 255,255,255,136,6,129,136,4,189,
   136,4,129,255,255,255,227 :rem 207
630 REM ** GEOWRITE ** :rem 139
640 DATA 255,255,255,146,73,1,255,255,125
   ,170,170,171,255,255,255,128:rem 90
650 DATA 0,1,129,94,129,129,170,129,128,0
   ,1,133,251,117,128,0 :rem 174
660 DATA 1,191,215,221,128,0,1,190,237,18
   1,128,0,1,189,181,253 :rem 225
670 DATA 128,0,1,187,124,1,128,0,1,134,24
   5,189,255,255,255,35 :rem 188

```

```

680 REM ** GEOWRITE APPL. #1** :rem 63
690 DATA 255,255,255,216,0,31,252,0,31,12
    9,188,31,129,94,1,129 :rem 242
700 DATA 235,1,128,0,1,128,0,1,128,0,1,25
    5,255,255,128,0 :rem 163
710 DATA 1,255,255,255,224,0,7,159,0,249,
    128,255,1,129,0,129 :rem 136
720 DATA 134,0,97,136,0,17,176,0,13,192,0
    ,3,255,255,255,98 :rem 39
730 REM ** GEOWRITE APPL. #2** :rem 60
740 DATA 31,255,248,64,0,2,0,15,0,128,127
    ,129,131,255,1,143 :rem 68
750 DATA 254,1,176,127,65,128,254,177,131
    ,184,1,142,16,1,128,12 :rem 30
760 DATA 1,143,255,241,128,0,1,136,146,12
    9,141,170,129,138,186,129 :rem 181
770 DATA 136,170,241,128,0,1,15,255,240,6
    4,0,2,31,255,248,251 :rem 171
780 REM ** GEOPAINT APPL.** :rem 221
790 DATA 255,255,255,128,0,1,145,17,1,185
    ,17,1,190,170,253,184 :rem 241
800 DATA 68,1,144,68,1,144,0,1,144,80,1,1
    44,80,1,184,80 :rem 128
810 DATA 1,191,223,253,184,80,1,144,80,1,
    144,80,1,144,3,255 :rem 70
820 DATA 144,2,1,144,2,245,144,2,189,128,2
    ,1,255,255,255,34 :rem 80

```



È Jackson

la biblioteca che fa testo
in Informatica

CONCETTI GENERALI

F. Filippazzi - G. Occhini

VOI E L'INFORMATICA
100 TAVOLE PER IL MANAGER
Cod. 526A 120 pag. L. 15.000

M. Langfelder - G. Occhini
VOI, L'AUTOMAZIONE E L'UFFICIO:
100 TAVOLE PER IL MANAGER
Cod. 545P 192 pag. L. 45.000

AA. VV.

INFORMATICA DI BASE
I CONCETTI FONDAMENTALI
HARDWARE E SOFTWARE
Cod. 158EC 240 pag. L. 55.000

N. Barcellona - A. Marini
I TERMINI DELL'INFORMATICA
E DELLE DISCIPLINE CONNESSE
Cod. 101H 464 pag. L. 50.000

J. Kerridge - N. Wills
ARCHITETTURE DI SISTEMA
Cod. GYS266 212 pag. L. 32.000

R. Doretti
DATA BASE:
CONCETTI E DISEGNO
Cod. 526P 192 pag. L. 22.500

L. Saret
DATA PROCESSING
Cod. GYS248 344 pag. L. 45.000

P. Bishop
CONCETTI DI INFORMATICA
Cod. GYS245 556 pag. L. 43.000

O. Carlon
I FONDAMENTI DELL'INFORMATICA
DAI SISTEMI DI CALCOLO AI
SISTEMI PER L'ELABORAZIONE
E LA TRASMISSIONE DATI
Cod. C1276 pag. L. 60.000

Puoi trovare i libri Jackson nelle migliori
librerie, oppure acquistarli con questo coupon.

Sono abbonato a _____
ed ho diritto allo sconto del 20% sino al 28/02/87
Se siete interessati al catalogo o all'acquisto
di alcuni libri potete ritagliare la cedola e
inviarla a:
Gruppo Editoriale Jackson S.p.A.,
Via Rosellini, 12 - 20124 Milano
(l'invio contrassegno sarà gravato da
L. 3000 di spese).
Catalogo ☐ Libro ☐

Titolo _____
Nome e Cognome _____
Via _____
Città _____ c.a.p. _____
Tel. _____

WORK

Multitasker

di R. Daley
trad. e adatt. di E. Stefani

Molti potenti computer, tra i quali il nuovo Amiga della Commodore, posseggono un grosso vantaggio rispetto al C64: la possibilità del multitasking, ovvero dell'esecuzione contemporanea di diversi programmi.

Ecco che questa volta SuperCommodore vi offre un'incredibile routine per dotare di questa caratteristica un'impareggiabile C64!

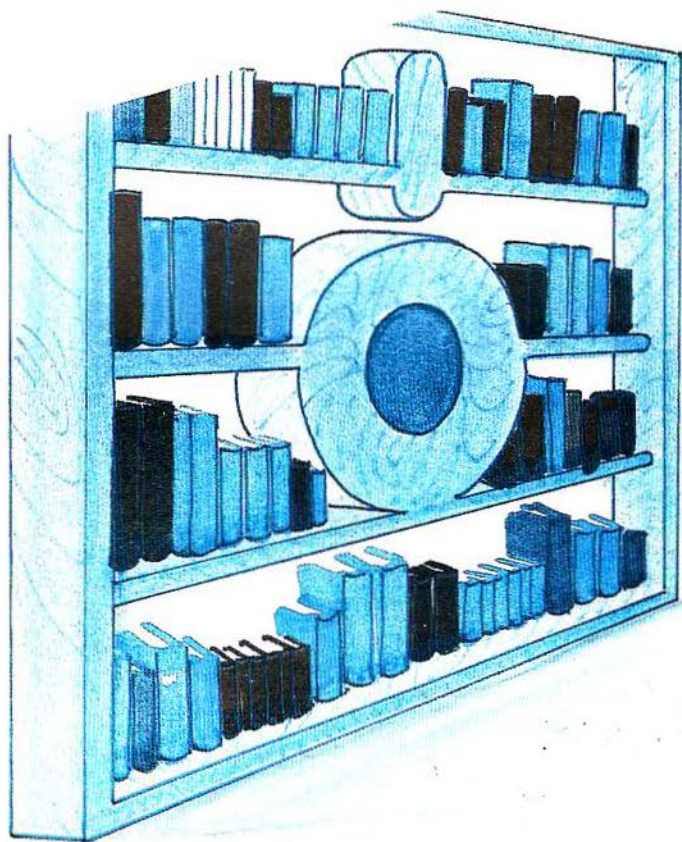
Sui numeri passati abbiamo già pubblicato una routine (Super Switcher) che permette di caricare contemporaneamente nella memoria del C64 diversi programmi: tre blocchi di memoria di 12 Kbyte possono contenere tre diversi programmi, e Super Switcher permette di trattare ogni blocco di memoria come un computer a sè stante.

Con Super Switcher tuttavia può essere eseguito soltanto un programma alla volta, pur essendocene in memoria più di uno.

Ora, con Multitasker, abbiamo la capacità di caricare in RAM due diversi programmi e quindi di mandarli in esecuzione entrambi, *contemporaneamente!*

Note per il caricamento e la digitazione di Multitasker

Il programma richiede per il caricamento la presenza dell'indirizzo secondario 1 nell'istruzione LOAD: si dovrà pertanto utilizzare la seguente procedura:



per il caricamento da cassetta, posizionarsi in corrispondenza dell'inizio del programma e digitare:

LOAD",1,1 (RETURN)

senza inserire spazi tra i doppi apici. Al termine del caricamento digitare rispettivamente:

NEW (RETURN)
SYS49152 (RETURN)

Il programma verrà in questo modo attivato. Si noti che è possibile ottenere un elenco completo di tutti i programmi presenti sulla cassetta e della loro posizione ponendo la cassetta completamente riavvolta nel registratore e digitando:

VERIFY *** (RETURN)

Poiché non esiste alcun programma dal nome ***, il computer visualizzerà sullo schermo i nomi di tutti i programmi incontrati (segnarsi la posizione del contastato sottraendo due o tre unità, per ogni apparizione del messaggio FOUND...) fino al termine del nastro.

Per il caricamento da disk drive, è possibile ottenere un elenco dei programmi presenti sul dischetto digitando:

```
LOAD"$",8 (RETURN)
LIST (RETURN)
```

Individuato il nome del programma, sarà sufficiente digitare:

```
LOAD"nome del programma",8,1
(RETURN)
```

e, al termine del caricamento, digitare:

```
NEW (RETURN)
SYS49152 (RETURN)
```

per ottenerne l'attivazione.

Coloro che invece devono digitare il programma copiando il listato dalla rivista, dovranno come sempre servirsi di MLX (vedi in altra parte della rivista) fornendogli i seguenti indirizzi:

indirizzo iniziale: **49152**
indirizzo finale: **49935**

una volta salvato il programma su nastro o disco, si potranno seguire per il suo caricamento e attivazione le istruzioni date più sopra.

I primi passi con Multitasker

Dopo aver dato la SYS 49152 di attivazione, sembrerà non accadere nulla tranne una breve attesa e la comparsa della scritta READY.

Premete il tasto F1 e digitate SYS 49155: questa volta lo schermo si cancellerà e comparirà la scritta MULTITASKER ENABLED sulla seconda linea dello schermo, per comunicare che il programma è pronto per funzionare.

La memoria è ora configurata in due blocchi indipendenti e separati (esattamente come se fossero due computer diversi) pronti per l'uso.

In questo momento vi trovate nel blocco 2: per averne conferma premete il tasto F7 e una scritta vi comunicherà il numero del blocco in cui vi trovate.

I blocchi 1 e 2 posseggono 17917 byte di memoria utilizzabili -verificato con PRINT FRE(0)-. Ritorniamo ora nel blocco 1 premendo il tasto F3: ci trovavamo infatti nel blocco 1 prima della SYS 49155. La pressione di F3 permette dunque di passare da un blocco all'altro.

Una nota molto importante: quando si

utilizza Multitasker non bisogna mai premere RUN/STOP e RESTORE, pena un blocco totale del computer.

Un semplice test

Se avete seguito la procedura descritta, vi trovate ora nel blocco 1: se invece siete nel blocco 2, premete F3 per passare al primo. Se non sapete in quale blocco vi trovate, premete F7. Una volta nel blocco 1, digitate il seguente programma:

```
100 FOR I=0 TO 10000
110 PRINT I
120 NEXT I
```

Passate quindi nel blocco 2 premendo F3 e digitate quest'altro programmino:

```
100 FOR I=10000 TO 0 STEP -1
110 PRINT ,I
120 NEXT I
```

Potete controllare come entrambi i programmi si trovino contemporaneamente in memoria premendo più volte F3 e digitando LIST.

Andate ora nel blocco 1 e date il RUN al programma: vedrete una colonna di numeri nella parte sinistra dello schermo, il cui valore cresce progressivamente da 0 a 10000. Osservatela per qualche secondo, quindi premete F3 per passare al blocco 2 e date il RUN: vedrete una colonna di numeri decrescenti da 10000 a 0; a causa della presenza della virgola nell'istruzione PRINT, la colonna risulta più centrale della precedente.

Osservate per qualche secondo l'esecuzione del programma, e premete ancora F3: questa volta vi troverete a osservare l'esecuzione del primo programma.

Pur essendo in esecuzione entrambi i programmi contemporaneamente, non si tratta ancora di vero multitasking: se osservate attentamente i numeri visualizzati quando passate da un blocco all'altro premendo F3, vi accorgete infatti che i programmi sono realmente in esecuzione soltanto quando selezionate il blocco di memoria cui appartengono.

Il vero multitasking si avrebbe se voi continuaste a premere ininterrottamente e molto velocemente il tasto F3.

Naturalmente non pretendiamo di fare eseguire a voi personalmente parte delle operazioni necessarie a ottenere un vero multitasking! Per questo esiste il tasto F5: la sua pressione provoca l'inizio del vero multitasking mandando in esecuzione contemporanea entrambi i programmi.

Si noti che la velocità di esecuzione di questi ultimi si è ridotta di circa la metà: pur potendo osservare la visualizzazione delle istruzioni di stampa di un programma alla volta, all'interno del C64 sono entrambi in esecuzione.

Potete facilmente constatarlo osservan-

do lo sviluppo della sequenza di numeri nei due blocchi premendo F3.

Premendo ancora F5 si interrompe il multitasking ritornando alle condizioni precedenti.

La seguente tabella riassume le operazioni dei tasti funzione, unitamente al valore che, inserito nella locazione 725 tramite una POKE, può simulare la pressione nel caso si desideri utilizzare da programma le capacità di Multitasker.

TASTO POKE AZIONE

F1	4	necessario solo in fase d'installazione
F3	5	passa tra i blocchi in modo standard e tra i due schermi in modo multitasking
F5	6	attiva e disattiva il modo multitasking
F7	3	visualizza il numero del blocco in uso

Le possibili applicazioni di Multitasker

Probabilmente avrete già delle idee per possibili applicazioni di questa preziosa routine; ve ne suggeriamo tuttavia qualcuna.

L'elaborazione "in sottofondo" è una possibilità attraente: se per esempio dovete effettuare un lungo ordinamento di stringhe, ponetelo in un blocco e mandatelo in esecuzione. Potrete intanto passare al blocco libero ed effettuare operazioni come l'edit o la scrittura di un programma, la sua esecuzione, la stampa di un listato su stampante, oppure più semplicemente farvi un bel videogame rilassante...

E' possibile inventare tutto un nuovo mondo per i videogiochi: creare dei programmi che facciano uso delle capacità di multitasker e combattano quindi tra di loro (un programma scritto da voi potrebbe scontrarsi con un programma scritto da un amico: il vincitore sarà chi avrà creato il programma in grado di maneggiare più velocemente e in modo più preciso il movimento della propria astronave e la traiettoria dei proiettili che distruggono quella dell'avversario!)

Alcune limitazioni

Data la non eccessiva memoria di cui è dotato il C64, i due blocchi non sono totalmente indipendenti, ma condividono alcune aree di memoria come per esempio la memoria colore e la memoria per l'INPUT/OUTPUT.

Altre parti comuni sono lo schermo ad alta risoluzione, i caratteri programmabili,

e gli effetti sonori. Gli SPRITE sono invece gestibili separatamente da ciascun blocco.

Multitasker utilizza i tasti funzione F1, F3, F5 ed F7: non è quindi compatibile

con programmi che utilizzino gli stessi tasti funzione.

Poiché le routine in linguaggio macchina di Multitasker occupa le locazioni a parti-

re dalla 49152, il suo utilizzo non è ovviamente compatibile con i programmi che facciano uso della stessa area di memoria.

Multitasker

49152	:076,089,192,120,169,008,142	49458	:076,091,193,076,249,193,160
49158	:141,136,002,169,000,141,083	49464	:165,197,141,237,194,169,135
49164	:000,082,141,001,082,141,203	49470	:000,141,213,002,108,246,004
49170	:002,082,170,189,223,194,110	49476	:194,032,112,194,076,091,255
49176	:149,043,232,224,014,144,062	49482	:193,173,244,194,073,255,182
49182	:246,169,002,141,255,194,013	49488	:141,244,194,173,255,194,001
49188	:141,000,195,169,037,141,207	49494	:205,000,195,240,221,169,092
49194	:024,208,169,066,133,251,125	49500	:001,141,241,194,165,197,007
49200	:169,192,133,252,160,000,186	49506	:141,197,152,141,197,156,058
49206	:177,251,240,006,032,210,202	49512	:173,255,194,201,001,240,144
49212	:255,200,208,246,088,096,129	49518	:007,201,002,240,072,076,196
49218	:147,013,077,085,076,084,036	49524	:056,193,186,142,214,002,141
49224	:073,084,065,083,075,069,009	49530	:169,255,141,250,194,162,013
49230	:082,032,069,078,065,066,214	49536	:000,160,152,032,050,194,204
49236	:076,069,068,013,000,120,174	49542	:173,241,194,240,045,174,177
49242	:173,020,003,141,246,194,099	49548	:253,194,142,177,194,142,218
49248	:169,236,141,020,003,173,070	49554	:189,194,232,232,142,185,040
49254	:021,003,141,247,194,169,109	49560	:194,169,002,141,255,194,083
49260	:192,141,021,003,169,032,154	49566	:169,000,141,250,194,169,057
49266	:141,240,194,169,255,141,230	49572	:180,141,251,194,169,193,012
49272	:250,194,162,160,160,160,182	49578	:141,252,194,162,156,160,211
49278	:032,050,194,162,224,160,180	49584	:000,076,050,194,174,214,116
49284	:224,032,050,194,173,040,077	49590	:002,154,076,056,193,186,081
49290	:003,141,248,194,169,193,062	49596	:142,214,002,169,255,141,087
49296	:141,040,003,173,041,003,033	49602	:250,194,162,000,160,156,092
49302	:141,249,194,169,194,141,214	49608	:032,050,194,173,241,194,060
49308	:041,003,169,053,133,001,044	49614	:240,232,174,254,194,142,162
49314	:169,076,141,052,235,169,236	49620	:177,194,142,189,194,232,060
49320	:174,141,053,235,169,194,110	49626	:232,142,185,194,169,001,117
49326	:141,054,235,169,134,141,024	49632	:141,255,194,169,000,141,100
49332	:219,228,169,002,141,220,135	49638	:250,194,169,180,141,251,135
49338	:228,173,243,194,141,242,127	49644	:194,169,193,141,252,194,099
49344	:194,169,004,141,240,194,110	49650	:162,152,160,000,076,050,074
49350	:169,001,141,000,195,141,077	49656	:194,169,001,133,251,169,141
49356	:255,194,169,000,141,237,176	49662	:195,133,252,164,211,177,106
49362	:194,141,213,002,141,000,133	49668	:251,240,005,145,209,200,030
49368	:012,141,001,012,141,002,013	49674	:208,247,173,255,194,024,087
49374	:012,170,189,209,194,149,121	49680	:105,048,145,209,230,214,199
49380	:043,232,224,014,144,246,107	49686	:165,214,201,024,176,018,052
49386	:088,096,173,213,002,208,246	49692	:165,209,024,105,040,133,192
49392	:033,169,000,141,241,194,250	49698	:209,144,002,230,210,169,230
49398	:173,237,194,201,064,208,043	49704	:000,133,211,076,056,193,197
49404	:008,169,000,141,239,194,235	49710	:198,214,208,245,134,252,017
49410	:076,034,193,173,239,194,143	49716	:142,252,152,142,252,156,124
49416	:208,024,169,001,141,239,022	49722	:132,254,140,254,152,140,106
49422	:194,173,237,194,201,004,249	49728	:254,156,174,240,194,160,218
49428	:240,082,201,005,240,043,063	49734	:000,132,251,140,251,152,228
49434	:201,006,240,045,201,003,210	49740	:140,251,156,132,253,140,124
49440	:240,019,173,244,194,240,118	49746	:253,152,140,253,156,177,189
49446	:017,206,242,194,208,012,149	49752	:251,145,253,200,208,249,114
49452	:173,243,194,141,242,194,207	49758	:238,252,000,238,254,000,052

49764 : 202,208,240,173,250,194,087
 49770 : 240,001,096,108,251,194,228
 49776 : 173,000,195,201,001,240,154
 49782 : 023,169,001,141,000,195,135
 49788 : 169,016,141,245,194,169,034
 49794 : 152,141,253,194,169,000,015
 49800 : 141,254,194,076,162,194,133
 49806 : 169,002,141,000,195,169,050
 49812 : 032,141,245,194,169,000,161
 49818 : 141,253,194,169,156,141,184
 49824 : 254,194,173,024,208,041,030
 49830 : 015,013,245,194,141,024,030
 49836 : 208,096,138,174,198,000,218
 49842 : 236,137,002,176,007,157,125
 49848 : 119,002,232,142,198,000,109
 49854 : 076,066,235,173,255,194,165
 49860 : 205,000,195,208,003,108,147
 49866 : 248,194,169,255,201,127,116
 49872 : 096,001,012,003,012,003,079
 49878 : 012,003,012,000,082,000,067
 49884 : 082,000,082,001,082,003,214
 49890 : 082,003,082,003,082,000,222
 49896 : 152,000,152,000,152,000,176
 49902 : 000,000,004,000,000,003,245
 49908 : 000,016,170,170,170,170,172
 49914 : 000,170,170,002,002,001,083
 49920 : 001,021,019,009,014,007,071
 49926 : 032,002,012,015,003,011,081
 49932 : 032,035,000,000,255,255,077



È Jackson la biblioteca che fa testo in Informatica

SISTEMI OPERATIVI MS-DOS

C. De Voney

MS-DOS LA GRANDE GUIDA
Cod. GY273 334 pag. L. 45.000

L. Suglia

**MS-DOS E PC-DOS
LO STANDARD IBM**
Cod. 094D 128 pag. L. 7.000

V. King - D. Waller

PC-DOS
Cod. D12H 64 pag. L. 8.500

V. King - D. Waller

MS-DOS
Cod. D19H 56 pag. L. 8.500

UNIX

M. Matteuzzi - P. Pellizzardi

AMBIENTE UNIX
Cod. 543P 200 pag. L. 19.000

R. Thomas - J. Yates

UNIX LA GRANDE GUIDA
Cod. G223 112 pag. L. 70.000

L. Blackburn - M. Taylor

UNIX
Cod. 009H 68 pag. L. 8.500

CP/M

R. Zaks

CP/M CON MP/M
Cod. 510P 320 pag. L. 29.000

E. Napoli

**CP/M
IL "SOFTWARE BUS"**
Cod. 093D 112 pag. L. 7.000

L. Blackburn - M. Taylor

CP/M
Cod. 011H 72 pag. L. 8.500

Puoi trovare i libri Jackson nelle migliori
librerie, oppure acquistarti con questo coupon.

Sono abbonato a _____
ed ho diritto allo sconto del 20% sino al 28/02/87.
Se siete interessati al catalogo o all'acquisto
di alcuni libri potete ritagliare la cedola e
inviarla a:

Gruppo Editoriale Jackson S.p.A.,
Via Rosellini, 12 - 20124 Milano
(l'invio contrassegno sarà gravato da
L. 3000 di spese).
Catalogo ☐ Libro ☐

Titolo _____
Nome e Cognome _____
Via _____
Città _____ c.a.p. _____
Tel. _____

WORK

Basic examiner

di R. Kephart
trad. e adatt. di S. Colombo

Se desiderate imparare come il vostro C64 memorizza i programmi BASIC, questa è l'utilità che fa per voi: può funzionare con qualsiasi programma BASIC ed offre un'analisi completa della sua struttura, esattamente come questa viene "vista" dal computer.

Che cosa succede all'interno del computer quando si digita un programma BASIC e lo si manda in esecuzione? Il C64 esegue in brevissimo tempo un numero enorme di operazioni diverse, internamente e senza mostrarlo all'operatore. Un programma BASIC, una volta immagazzinato all'interno della memoria del computer, assume un aspetto alquanto differente da quanto ci si possa aspettare. In primo luogo, tutte le istruzioni (per esempio FOR, NEXT, GOTO e così via) vengono "tokenizzate".

Ciò significa che ogni istruzione appartenente al "vocabolario" del BASIC viene convertita in un singolo numero non appena la linea di istruzioni è memorizzata con la pressione di RETURN. Questa tecnica riduce l'ammontare di memoria occupata dal programma incrementandone nel contempo la velocità di esecuzione.

I numeri di linea assegnati vengono convertiti in un formato costituito da due byte: per esempio il numero 100 viene



convertito in uno 0 e un 100 ($0 \times 256 + 100 = 100$) mentre il numero 1000 viene convertito in un 3 e un 232 ($3 \times 256 + 232 = 1000$).

Nel momento in cui una linea viene inserita in memoria, il computer aggiunge uno zero come ultimo byte della linea stessa, per identificarne il punto finale. Analogamente, il punto finale dell'intero programma è identificato da tre zeri con-

tigui che vengono automaticamente aggiunti dall'interprete BASIC.

Altri valori numerici aggiunti dall'interprete sono i cosiddetti "link di linea": numeri di due byte che indicano la locazione di memoria in cui inizia la linea successiva.

Ogni cosa viene immagazzinata nella memoria del C64 in locazioni da 0 a 65535: normalmente un programma BA-

SIC inizia nella locazione 2049, e può terminare in una locazione compresa tra 2049 (nessun programma BASIC) e 40959 (massima lunghezza disponibile).

Un ipotetico programma BASIC che iniziassi nella locazione 2049 e terminasse nella 40959 potrebbe essere caricato in memoria ma non potrebbe mai essere eseguito: mancherebbe infatti lo spazio sufficiente - nella memoria BASIC - per ospitare le variabili, i vettori, e tutti i parametri necessari alla sua esecuzione.

Ogni carattere costituente il programma che non sia né un numero di linea né un'istruzione (per esempio l'argomento di 10PRINT "CIAO") viene immagazzinato in memoria come valore ASCII. Per esempio, se si digita la lettera A, essa viene memorizzata come un 65: PRINT CHR\$(65) visualizzerà infatti la lettera A.

Per chi utilizza il nastro o il disco

Chi utilizza il disk drive dovrà digitare la seguente linea di programma:

```
10 POKE44,192:POKE56,208:POKE
49152,0:LOAD "BASIC EXAMINER",8
```

e salvarla su disco come programma a sé stante, magari con il nome di "LOADER". Invece di caricare direttamente "BASIC EXAMINER", caricate "LOADER" e date il RUN: esso provvederà a

caricare e attivare automaticamente il programma vero e proprio.

La funzione del programma che abbiamo deciso di chiamare "LOADER" è quella di spostare il programma principale in un'area di memoria normalmente non utilizzata dal BASIC; tale area inizia con la locazione 49152. Questo metodo permette di lasciare libero lo spazio normalmente dedicato a ospitare i programmi BASIC in modo da poter caricare tranquillamente il programma da analizzare.

Chi invece utilizza il registratore a cassette dovrà, prima di caricare "BASIC EXAMINER", dare in modo diretto la seguente serie di istruzioni:

```
POKE44,192:POKE56,208:POKE
49152,0:NEW
```

Solo a questo punto si potrà procedere al caricamento da cassetta del programma.

Come funziona

Per prima cosa, BASIC Examiner vi chiede il nome del programma da analizzare: se avete il disk drive, digitate il nome di un programma BASIC presente sul disco stesso. Se invece possedete il registratore, digitate il nome di un programma BASIC che possedete su un'altra cassetta e inserite quest'ultima nel registratore.

La seconda domanda rivolta dal programma riguarda il numero di linea dal

quale si vuole iniziare l'esplorazione; premendo RETURN si analizzerà il programma partendo dalla sua prima linea di istruzioni.

Ogni locazione di memoria utilizzata dal programma analizzato viene visualizzata secondo questo schema:

- il numero della locazione di memoria

- il numero contenuto nella locazione, in forma sia decimale che esadecimale. Se il numero rappresenta un'istruzione BASIC oppure un indicatore di fine linea, è evidenziato.

- se il contenuto della locazione non rappresenta un'istruzione BASIC, viene visualizzata una stringa rappresentante il suo valore ASCII.

- il token associato al numero contenuto nella locazione, oppure il suo valore ASCII se significativo (per esempio se il numero provoca un cambiamento di colore o una pulizia dello schermo, eccetera.). Viene inoltre specificato se il numero rappresenta un link di linea oppure un numero di linea.

E' possibile rallentare la visualizzazione dei valori tenendo premuto il tasto CTRL, oppure fermarla temporaneamente tenendo premuto il tasto SHIFT. La linea più alta dello schermo non è soggetta a scorrimenti e contiene informazioni generali sul programma in esame.

BASIC Examiner

```
100 IFPEEK(46)<PEEK(44) THEN POKE45,PEEK(17
4):POKE46,PEEK(175):CLR :rem 76
110 GOSUB360:SYS844:POKE808,239:PRINT"
{CLR}{WHT}" :rem 250
120 POKE648,160:PRINT"{HOME}{RVS}LOCAZ.
{ 2 SPAZI}{ 1 SH B}HEX{ 1 SH B}DECIM.
{ 1 SH B}SIGNIFICATO{ 8 SPAZI}":POKE
648,4 :rem 84
130 D=256:POKEA,FNH(S):POKEX,FNL(S):SYS82
8:M=PEEK(A)+D*PEEK(X):D=16:PRINT
:rem 31
140 FORI=33TO34:V(0)=PEEK(M):V(1)=PEEK(M+
1):IFV(0)+V(1)=0 THEN NEXT:GOTO330
:rem 120
150 FORJ=0TO1:V=V(J):PRINTM+JTAB(9);
:rem 235
160 PRINTTAB(9)CHR$(FNC(FNH(V)))CHR$(FNC(
FNL(V)))TAB(14)V(J):IFJ=0 THEN PRINT
:rem 120
170 NEXT:M=M+2:PRINTTAB(21)S$(I)V(0)+256*
V(1):PRINT:NEXT:M=M-1 :rem 248
180 M=M+1:V=PEEK(M):WAIT653,1,1:IFPEEK(20
3)=63 THEN 350 :rem 100
```

```
190 B=(V>127ANDV<204ORV=255)ANDNOTQ:IFVTH
EN210 :rem 89
200 PRINTMTAB(9)"{RVS}00"TAB(15)"0{OFF}"T
AB(21)"{RVS}FINE LINEA{GIU}":M=M+1:Q
=V:GOTO140 :rem 39
210 PRINTMTAB(9)CHR$(18*-B)CHR$(FNC(FNH(V
)))CHR$(FNC(FNL(V)))"0{OFF}"TAB(15);
:rem 191
220 PRINTCHR$(18*-B)MID$(STR$(V),2)"{OFF}
"TAB(22); :rem 131
230 IFV=32 THEN PRINT"{ 2 SPAZI}"S$(V):GOTO
180 :rem 70
240 IFV=34 THEN Q=NOTQ:PRINTCHR$(V):GOTO180
:rem 153
250 IFV=141 AND NOT B THEN PRINT"{RVS}{ 1 SH M
{OFF}"S$(V):GOTO180 :rem 240
260 IFV=255 THEN PRINT" [<>]":GOTO180
:rem 97
270 IFB THEN PRINT" ";:POKEA,V:SYS936:PRINT
:GOTO180 :rem 127
280 IFV>32 THEN 310 :rem 229
290 PRINT"{RVS}CHR$(V+64)"0{OFF}"S$(V);:
IF$(V)="" THEN PRINT"[CTRL-CHR$(V+64)
]"; :rem 40
```

```

300 PRINT:GOTO180                                :rem 44
310 IFV<128ORV>160THENPRINTCHR$(V):GOTO180      :rem 140
0                                                :rem 140
320 PRINTQ$D$CHR$(V)Q$D$TAB(24)S$(V):GOTO180     :rem 60
330 M=M-1:PRINT"{ 2 SU}";:POKEX,PEEK(214)        :rem 227
:SYS59903:PRINT:FORI=0TO2
340 PRINTM+ITAB(9)"00"TAB(15)"0":NEXT:PRI        :rem 211
NT"{SU}"TAB(22)"{CYN}FINE PROGRAMMA
[<8>]"
350 POKE198,0:GOSUB400:PRINT"{CLR}":GOTO130      :rem 184
30
360 I$="{CLR}{ 2 GIU' }{WHT}BASIC EXAMINER      :rem 1'86
":DIMS$(160):Q$=CHR$(34):D$=CHR$(20):
A=780:X=A+1
370 Y=A+2:DEFFNH(F)=INT(F/D):DEFFNL(F)=F-D*FNH(F):DEFFNC(F)=F+48-7*(F>9)
D*FNH(F):DEFFNC(F)=F+48-7*(F>9)
380 FORI=5TO34:READS$(I):NEXT:FORI=129TO160:READS$(I):NEXT
390 FORI=828TO931:READJ:POKEI,J:NEXT
400 F$="":Q=0:IFPEEK(788)=49THENPRINTI$:INPUT"{GIU'}NOME PROGRAMMA":F$
410 F=LEN(F$):S$="":INPUT"{GIU'}INIZIO CO N LINEA ":S$:S=INT(VAL(S$)):IFF=0THEN
RETURN
420 FORI=1TOF:POKE678+I,ASC(MID$(F$,I,1)):NEXT:POKE183,F:POKE157,128
430 FORI=184TO188:READJ:POKEI,J:NEXT:POKEA,0:POKEX,1:POKEY,8:SYS65493:rem 180
440 POKE44,8:SYS42291:POKE44,192:IFST=64THENRETURN
450 PRINT:IFST<0THENPOKEX,5:SYS42039

```

```

460 OPEN15,8,15:INPUT#15,A,B$:CLOSE15:PRI
NT"{GIU'}{RVS}"B$
470 DATA BIANCO,,,LOCK,UNLOCK,,,RETURN,M
NUSCOLO,,
480 DATA CRSR GIU',REV-ON,HOME,DEL,,,,,
ROSSO,CRSR DES.,VERDE,BLU,SPAZIO
490 DATA PROSS.LINEA A,NUMERO LINEA
500 DATA ARANCIO,,,F1,F3,F5,F7,F2,F4,F6,
F8,SHIFT-RETURN,MAIUSCOLO,,NERO
510 DATA CRSR SU,RVS-OFF,CLEAR,INST,MARRO
NE,ROSSO CH,GR.1,GR.2,VERDE CH,BLU CH
520 DATA GR3,VIOLA,CRSR SIN.,GIALLO,CIANO
,SHIFT-SPAZIO
530 DATA 134,20,133,21,169,1,162,8,32,23,
166,165,95,166,96,96,120,169,127,141
540 DATA 13,220,169,3,141,21,3,169,120,14
1,20,3,169,129,141,26,208,169,27,141
550 DATA 17,208,88,160,31,185,35,167,153,
167,3,136,208,247,96,41,127,76,210
560 DATA 255,169,1,141,25,208,162,250,160
,199,169,21,44,18,208,16,6,162,57,160
570 DATA 197,169,133,142,18,208,140,0,221
,141,24,208,173,13,220,41,1,240,3,76
580 DATA 49,234,76,188,254,0,0,8,167,2

```

CEDOLA DI COMMISSIONE LIBRARIA

VOGLIATE SPEDIRMI

n° copie	codice	Titolo	Prezzo unitario	Prezzo totale
Totale				

☐ Pagherò contrassegno al postino il prezzo indicato più **L. 3.000 per contributo fisso spese di spedizione.**

Condizioni di pagamento con esecuzione del contributo spese di spedizione:

☐ Allego assegno della Banca ☐ Allego fotocopia del versamento sul c/c n. 11666203 a voi intestato

N° ☐ Allego fotocopia di versamento su vaglia postale e voi intestato

Nome e Cognome

Via

Cap

Città

Prov.

Data

Firma

Spazio riservato alle Aziende Si richiede l'emissione di fattura

ORDINE
MINIMO
L. 50.000

In busta chiusa inviate
questo coupon a:
Gruppo Editoriale Jackson
via Rosellini, 12 - 20124 MI

Partita I.V.A.

WORK

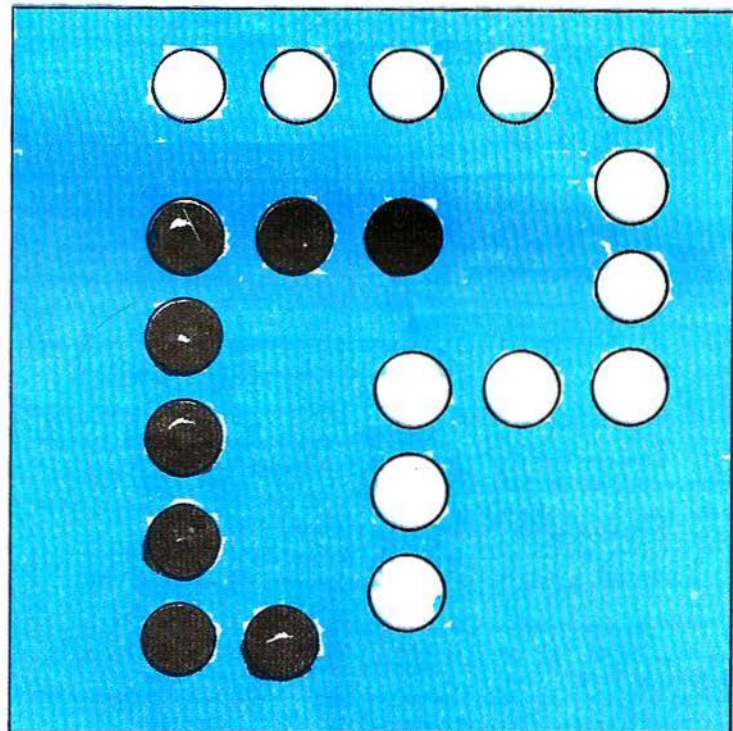
MetaBasic plus per C64

di K.Mykytyn

trad. e adatt. di M.Cristuib Grizzi

Ecco MetaBasic Plus, una versione completamente aggiornata del nostro famoso programma MetaBasic: con molti nuovi comandi rispetto alla versione precedente, MetaBasic Plus è una poderosa utility che cambierà il vostro modo di programmare. Aggiunge 43 nuovi comandi al BASIC del vostro C64, indirizzati principalmente al 'debugging' e al controllo dei vostri programmi; è utilizzabile sia da sola, sia in congiunzione con un monitor/assembler di linguaggio macchina come il SUPERMONITOR da noi già pubblicato. Sul prossimo numero troverete la versione di MetaBasic per il C128.

Avete acquistato la vostra prima automobile e vi trovate molto bene ma, la prima volta che entrate in autostrada, vi accorgete con dispiacere che non superate i 100 chilometri all'ora. Che fare? Portatela dal vostro meccanico di fiducia, ed egli potrà fornirvi tre alternative: togliere il motore e sostituirlo con un altro; oppure aggiungere al motore che già avete qualche marchingegno turbocompresso che ne migliori la potenza, oppure ancora, senza aggiungere o togliere nulla, operare una buona messa a punto tramite un'apposita macchina che misura le prestazioni del motore. E' possibile aggiungere al proprio computer dei nuovi comandi di programmazione in tre modi simili: il primo è disabili-



tare completamente il BASIC e creare 'ex novo' un nuovo linguaggio (un nuovo motore) basato sulle vostre idee circa ciò che dovrebbe permettere un buon linguaggio di programmazione.

Il secondo metodo, un'estensione del linguaggio, mantiene il BASIC e aggiunge alcuni nuovi comandi di programmazione (per suono, grafica in alta risoluzione, o altre funzioni specializzate). Rimane il 'motore' BASIC, ma vengono aggiunti alcuni 'pezzi' che lo rendono più veloce e più efficiente. Il 'Simon's Basic' e la 'Super Expander 64' ne sono degli esempi.

Il terzo metodo è come una messa a punto che non modifica il motore. Si aggiungono comandi da usarsi solo in modo diretto per il 'debugging'. Non si tratta di un nuovo linguaggio o di un'estensione del BASIC, ma di un cosiddetto 'Sistema di sviluppo'. I nuovi comandi aggiunti non possono essere usati all'interno di un programma, in quanto funzionano soltanto in modo immediato.

I nuovi linguaggi e le estensioni offrono molteplici vantaggi, ma hanno tutti un grosso lato negativo: si è obbligati a caricare in memoria il linguaggio o l'estensione prima di caricare un programma

nel nuovo linguaggio, o che usi i comandi dell'estensione stessa: in caso contrario, non funzionerà nulla e si otterranno messaggi di 'SYNTAX ERROR' a valanga.

Il vantaggio di un sistema di sviluppo come MetaBasic Plus, risiede nel fatto che l'utilità può essere caricata al bisogno, quando si sta scrivendo o verificando un programma. Una volta che questo sarà finito e corretto, non si dovrà più caricare MetaBasic Plus per mandarlo in esecuzione.

Un'introduzione a MetaBasic Plus

MetaBasic Plus usa termini mnemonici, così non dovete ricordare una gran quantità di indirizzi di SYS. Inoltre, se dimenticate la sintassi dei nuovi comandi, potrete o rileggere questo articolo, o semplicemente digitare HELP sulla vostra tastiera.

I programmatori in BASIC disporranno di 12 nuovi comandi. Per la scrittura dei programmi sono disponibili AUTO, KEY, e UNNEW. Per esaminare e modificare i programmi, ci sono CHANGE, DELETE, FIND, RENUM e VCHANGE.

Appositamente per il 'debugging' sono previsti DUMP, SPEED, TRACE e TROFF. Se siete programmatori in ASSEMBLY, potete usare comandi come MEMORY, MONITOR, NUMBER, e @. Per controllare MetaBasic Plus stesso, disponete di DEFAULT, HELP, INT, e QUIT.

I comandi per la gestione del disk drive comprendono BSAVE, CAT, DLIST, ERR, MERGE, READ, RESAVE, SCRATCH, SEND, e START.

Infine, esiste LLIST se avete una stampante e TERMINAL se avete un modem.

I comandi di MetaBasic Plus

Ecco un elenco alfabetico dei nuovi comandi, e le istruzioni su come usarli comprendenti degli esempi. I comandi di MetaBasic Plus e le stringhe sono stampati in nero, mentre i numeri lo sono in corsivo. Tutto ciò che è racchiuso tra parentesi è opzionale.

Se qualcosa è descritto come comando per disk drive, non funzionerà se non avete collegato un disk drive. Inoltre, alcuni degli aiuti di programmazione per L/M (linguaggio macchina) possono essere anche utili in BASIC, e viceversa.

AUTO - Programmazione BASIC

Sintassi: **AUTO** *inizio, incremento*

AUTO può eliminare un'operazione molto noiosa quando si scrive un programma. Genera automaticamente i numeri di linea del programma, partendo dal numero dato come inizio, ed incrementandolo con il numero dato come incremento. Separate i due numeri con una virgo-

la. Dopo aver premuto RETURN alla fine di una linea, il prossimo numero di linea verrà automaticamente generato dal computer. Il numero di linea corrente può essere modificato con l'uso del tasto INST/DEL, e sostituito da un altro numero. Premere RUN/STOP per uscire dal modo AUTO.

Esempio:

AUTO 100,10 parte da 100 e continua con 110, 120, 130, etc.

BSAVE - Comando per disk drive (vedi anche RESAVE)

Sintassi: **BSAVE** "nome file",*ind.iniziale, ind.fine*+1

BSAVE (Binary SAVE) salva un'area di memoria su disco, dall'indirizzo iniziale all'indirizzo finale. Inserite il nome del file tra le virgolette, e usate virgole per separare nome, indirizzo iniziale, ed indirizzo finale. E molto importante aggiungere 1 al valore dell'indirizzo più alto che volete salvare.

Potete usare questo comando per creare copie di programmi in linguaggio macchina, sempre che ne conosciate gli indirizzi iniziale e finale. BSAVE può essere usato anche per salvare porzioni di memoria di schermo, set di caratteri ridefiniti, o schermate in alta risoluzione.

I numeri devono essere in decimale: se desiderate tradurre dei valori da esadecimale in decimale, vedere il comando NUMBER.

Dopo aver usato BSAVE, si può ricaricare il file con LOAD "nome file",8,1.

Esempio:

BSAVE "METABASIC",35145,40960 per creare su disco una copia di MetaBasic Plus stesso.

CAT - Comando per disk drive (vedi anche DLIST, READ)

Sintassi: **CAT**

Ogni volta che desiderate leggere l'intera directory del disco inserito nel drive, digitate questo comando. Il programma BASIC correntemente in memoria non verrà cancellato. Per leggere solo specifiche parti della directory, vedi DLIST.

CHANGE - Programmazione BASIC (vedi anche FIND, VCHANGE)

Sintassi: **CHANGE@VECCHIO@NUOVO@**(*inizio, fine*)

Oppure: **CHANGE@"VECCHIO"@ "NUOVO@"**(*inizio, fine*)

CHANGE cerca nel programma in memoria la stringa definita in 'VECCHIO', e la sostituisce con quella definita in 'NUOVO'. Le stringhe possono essere lunghe fino a 30 caratteri, e devono essere racchiuse tra due '@'. Tutte le linee in cui vengono operate le sostituzioni sono listate sullo schermo.

Il primo tipo di sintassi sostituirà istruzioni BASIC e nomi di variabili. Il secondo tipo di sintassi deve essere usato per sostituire le stringhe. I valori 'inizio'

e 'fine' indicano i numeri di linea da cui deve iniziare e finire la sostituzione; se omessi, questa viene effettuata su tutto il programma. Esempio:

CHANGE@X@QQ@1,200 sostituisce al nome di variabile X il nome QQ, in tutte le linee tra 1 e 200. Le rimanenti linee non vengono modificate.

Per cambiare la stringa "PIPO" in "PLUTO" su tutto il programma, si usa per esempio:

CHANGE@"PIPO"@ "PLUTO"@

CURSORFREE - Programmazione BASIC (Vedi anche CURSORPRINT)

Sintassi: **CURSORFREE**

Disabilita il "modo virgolette" fino a che venga dato il comando CURSORPRINT o fino a che vengano premuti i tasti RUN/STOP e RESTORE.

CURSORPRINT - Programmazione BASIC (Vedi anche CURSORFREE)

Sintassi: **CURSORPRINT**

Riabilita il "modo virgolette", invertendo l'effetto del comando CURSORFREE.

DEFAULT - Comando di MetaBasic Plus (vedi anche INT, QUIT)

Sintassi: **DEFAULT** *bordo, schermo, testo, #periferica*

Quando si preme RUN/STOP+RESTORE, lo schermo ritorna ai colori di default consistenti in caratteri azzurri su schermo blu con bordo azzurro, lo sfondo o no. Inoltre, molti comandi come LOAD e SAVE assumono come numero di periferica di default 1, il registratore a cassette. Il comando DEFAULT permette di modificare questi valori in altri che più vi aggradano.

Se usate abitualmente il disk drive, potete cambiare il numero di periferica di default da 1 in 8. Se desiderate usare più spesso il vostro secondo drive, potete usare come numero di periferica di default il 9. Se il vostro C64 è collegato ad un televisore in bianco e nero, può risultare utile modificare la combinazione di colori fondo/testo su valori più leggibili.

Nota: non è possibile usare nessuno dei nuovi comandi per il disk drive una volta posto a un valore diverso da 1 il numero di periferica di default. Per disabilitare DEFAULT e ritornare alle condizioni normali, usate il comando MONITOR descritto più avanti. Esempio:

DEFAULT1,1,0,8 regola il colore dello schermo e lo sfondo sul bianco, il colore del testo sul nero, e il numero di periferica di default su 8. Se premete RUN/STOP+RESTORE, vedrete caratteri neri su uno sfondo bianco, e potrete usare, per caricare e salvare su disco, le istruzioni LOAD "nome file" e SAVE "nome file" (senza cioè dover aggiungere il ,8).

DELETE - Programmazione BASIC

Sintassi: **DELETE** *inizio-fine*

DELETE cancella dalla memoria un certo numero di linee del vostro programma.
Separate il numero della prima linea da cancellare (inizio) dall'ultimo (fine) con il segno meno (-). Esempio:
DELETE200-250 cancella tutte le linee tra 200 e 250, estremi compresi.

DLIST - Comando per disk drive (vedi anche CAT, READ)

Sintassi: **DLIST** "nome file"

Questo comando lista sullo schermo un programma BASIC contenuto in un disco, senza disturbare ciò che è attualmente contenuto in memoria. Il nome del programma deve essere racchiuso tra virgolette. **DLIST** vi permette di controllare il listato di un programma prima di usare, per esempio, **MERGE** o **SCRATCH**.

Permette inoltre di leggere porzioni della directory. **DLIST** "\$A" visualizzerà sullo schermo il nome di tutti i file nella directory il cui nome inizia con la lettera A. Esempio:

DLIST "MAGIC HAND" lista sullo schermo il programma "Magic Hand" registrato su disco, senza cancellare il programma BASIC attualmente residente in memoria.

DUMP - Programmazione BASIC

Sintassi: **DUMP**

Potete usare questo comando per esaminare il contenuto di tutte le variabili usate da un programma, esclusi i vettori e le matrici. Date il **RUN** al programma, quindi premete **RUN/STOP** e digitate **DUMP**. Per riprendere l'esecuzione del programma, usate **CONT**.

END - Programmazione BASIC

Sintassi: **END**

Visualizza l'ultima locazione di memoria utilizzata dal programma BASIC contenuto in RAM.

ERR - Comando per disk drive

Sintassi: **ERR**

ERR legge il canale d'errore del drive: usatelo quando la luce rossa del drive sta lampeggiando.

FIND - Programmazione BASIC (vedi anche CHANGE, VCHANGE)

Sintassi: **FIND** @STRINGA@ (inizio, fine)

oppure: **FIND** @"STRINGA"@ (inizio, fine)

Vi permette di trovare una parola, un'istruzione BASIC, una variabile, all'interno di un programma. Ogni linea contenente la stringa specificata viene listata sullo schermo. Se desiderate operare la ricerca solo su una parte del programma, specificate in 'inizio' e 'fine' i numeri di linea da cui iniziare e finire. Se non specificati, la ricerca avviene su tutto il programma. Se volete ricercare istruzio-

ni BASIC, variabili, o numeri, usate la prima sintassi. Il secondo tipo di sintassi deve essere usato se si ricercano stringhe o caratteri racchiusi tra virgolette. Esempio:
FIND @A=@ ricerca tutte le linee in cui viene definita la variabile A.

FREEMEM - Programmazione BASIC

Sintassi: **FREEMEM**

Visualizza il numero di byte liberi senza l'effettuazione della "garbage collection".

HELP - Comando di MetaBasic Plus

Sintassi: **HELP**

Questo comando fornisce una lista completa sullo schermo di tutti i comandi di MetaBasic Plus.

INT - Comando di MetaBasic Plus (vedi anche DEFAULT, QUIT)

Sintassi: **INT**

Alcune delle opzioni di MetaBasic Plus sono gestite da interrupt. Se si reinizializzano gli interrupt (attraverso il comando **MONITOR**), non funzioneranno più i tasti funzione e il comando **SPEED**. **INT** ripristina la gestione degli interrupt nelle condizioni standard di MetaBasic Plus.

KEY - Programmazione BASIC (vedi anche INT)

Sintassi: **KEY** #funzione, "comando o stringa"

Questo comando fornisce a MetaBasic Plus una grande flessibilità, permettendo di definire ciascuno degli otto tasti funzione con un differente comando o stringa.

Il comando, che può essere lungo fino a dieci caratteri, deve essere racchiuso da virgolette. Esistono inoltre due caratteri speciali: la freccia a sinistra funziona come un **RETURN**, per fare in modo che non dobbiate anche premere **RETURN** dopo la visualizzazione del comando associato al tasto funzione. Inoltre, l'apostrofo (**SHIFT+7**) vale come virgolette.

Usando **KEY**, potete disporre di un gran numero di comandi con la semplice pressione di un tasto (come avviene nel BASIC 3.5 del C16-Plus/4).

Se gli interrupt vengono accidentalmente modificati, dovete usare il comando **INT** per riabilitare il comando **KEY**.

Esempio:

KEY1, "{CLR}LIST100-" pulisce lo schermo e lista il programma a partire dalla linea 100 ogni volta che premete **F1**. (la freccia a sinistra indica che il **RETURN** viene dato automaticamente).

KEY7, "DATA" può essere utile con la numerazione automatica (vedi **AUTO**), se si sta scrivendo un programma con molte linee DATA. Dopo aver inserito una linea, premete **RETURN** e verrà visualizzato il numero di linea successivo. Premete quindi **F7** ed apparirà automatica-

mente la parola DATA.

KEY2, "VERIFY" definisce **F2** in modo che stampi **VERIFY** più un **RETURN** (si noti che gli apostrofi sono stati cambiati in virgolette). Se avete usato **DEFAULT** per cambiare il numero di periferica di default in 8, la pressione di **F2** provocherà la verifica automatica del programma salvato più recentemente sul disco.

LLIST - Comando per la stampante

Sintassi: **LLIST** inizio-fine

Questo comando lista un programma, ma il listato viene fatto sulla stampante invece che sullo schermo. I numeri di linea 'inizio' e 'fine' sono opzionali. La sintassi di **LLIST** è identica a quella di **LIST**. Esempio:

LLIST 10-20 lista su stampante le linee di programma tra 10 e 20, estremi compresi.

MAKEDATA - Programmazione BASIC

Sintassi: **MAKEDATA** inizio, fine

Converte il contenuto dell'area di memoria dalla locazione 'inizio' alla locazione 'fine' in linee DATA. Questa istruzione cancella qualsiasi programma BASIC residente in memoria per sostituirlo con le linee DATA generate. **MAKEDATA** può convertire in linee DATA un numero massimo di 4352 byte, e le linee generate avranno numeri di linea partenti da zero con 17 byte per linea.

Dopo aver creato le linee DATA, queste possono essere rinumerate con il comando **RENUM**, oppure fuse con un altro programma attraverso il comando **MERGE**.

MEMORY - Programmazione L/M (vedi anche @)

Sintassi: **MEMORY** ind.iniziale(-ind.finale)

E' possibile esaminare, con questo comando, qualsiasi parte della memoria. Usate numeri decimali (non esadecimali) per gli indirizzi iniziale e finale. I valori contenuti in memoria vengono visualizzati, sei byte per linea, in decimale. Inoltre, l'equivalente in caratteri ASCII viene visualizzato in campo inverso nella parte destra dello schermo. Questo comando ha un effetto molto simile al comando **M** di **SUPERMONITOR**, con la differenza che **MEMORY** lavora in decimale. Se non esiste un corrispondente carattere ASCII, viene visualizzato un punto.

Se si omette l'indirizzo finale, ad esempio con **MEMORY43**, verrà visualizzato il contenuto di due byte (43 e 44). Ciò rende facile leggere il contenuto dei puntatori a due byte. Per modificare il contenuto della memoria, si può usare il comando **@** come descritto più avanti. Esempio:
MEMORY41374-41474 visualizza alcuni dei primi messaggi d'errore (si noti

che il valore ASCII dell'ultimo carattere è sempre sommato a 128). Oppure, caricate un programma BASIC ed usate: MEMORY2048-2148 per vedere come quest'ultimo è immagazzinato in memoria.

MERGE Comando per disk drive

Sintassi: **MERGE** "nome file"

MERGE legge un programma da disco, lista ogni sua linea sullo schermo, e la aggiunge al programma attualmente in memoria. Se i due programmi possiedono numeri di linea comuni, il programma su disco ha la precedenza. Poniamo che entrambi i programmi posseggano la linea 250: la linea 250 del programma su disco rimpiazzerà la linea 250 del programma in memoria.

Prima di usare questo comando, può essere utile usare DLIST per assicurarsi di operare il MERGE del programma correttamente. Se ci sono conflitti nei numeri di linea, si può usare RENUM (vedi più avanti) per rinumerare uno dei due programmi. Se desiderate fare il MERGE di solo una parte di programma, usate DELETE per cancellare la parte che non interessa.

MONITOR - Programmazione L/M

Sintassi: **MONITOR**

Se avete in memoria un monitor in linguaggio macchina (per esempio il SUPERMONITOR), potete entrarvi con questo comando (ammesso che esso venga abilitato tramite un BRK). Per usare MetaBasic Plus con un monitor, dovete caricare MetaBasic Plus, digitare NEW, e dare la SYS36864. Quindi, caricare il monitor, digitare NEW, e dare la SYS di attivazione del monitor (49152 per il SUPERMONITOR) che inizierà il vettore BRK facendo in modo che punti al monitor.

MONITOR serve anche ad altre operazioni: riporta i colori di testo, bordo e schermo ai valori di default (blu ed azzurro), e riporta allo standard l'interrupt, disabilitando quindi le definizioni dei tasti funzione e il comando SPEED. Essi si possono comunque ripristinare con il comando INT.

NUMBER - Programmazione L/M

Sintassi: **NUMBER**\$esadecimale

oppure: **NUMBER**-decimale

NUMBER permette di convertire numeri da base decimale in esadecimale, e viceversa. I numeri esadecimali devono essere preceduti dal segno \$. Inoltre, il numero viene convertito nel formato byte basso / byte alto (in decimale) e nel numero binario equivalente (contrassegnato dal segno %). Esempio:

NUMBER\$100

256

01

%100000000

NUMBER34

\$22

340

%100010

POINTER - Programmazione BASIC

Sintassi: **POINTER**indirizzo

Questo comando è molto utile per il controllo dei puntatori del BASIC: effettua l'operazione equivalente a una PRINT PEEK(indirizzo)+256*PEEK(indirizzo+1).

QUIT - Comando di MetaBasic Plus (Vedi anche RESET)

Sintassi: **QUIT**

Reinizializza tutti i vettori e disabilita tutti i comandi di MetaBasic Plus. Non ripristina tuttavia il puntatore di fine memoria, lasciando MetaBasic Plus ancora protetto dal BASIC. Si può riattivare MetaBasic Plus con SYS36864, oppure con SYS9*4096.

READ - Comando per disk drive (vedi anche CAT, DLIST)

Sintassi: **READ** "nome file seq"

READ permette di esaminare il contenuto di file sequenziali su disco. Il contenuto del file viene visualizzato sullo schermo senza alterare il contenuto della memoria. Nel raro caso in cui vorreste usare il comando BASIC READ (per leggere istruzioni DATA) in modo diretto, dovreste farlo precedere dai due punti.

RENUM - Programmazione BASIC

Sintassi: **RENUM**(inizio),(incremento)

L'intero programma BASIC in memoria viene rinumerato, partendo dal numero di linea eventualmente specificato in 'inizio', con incremento eventualmente specificato in 'incremento'. Se si omettono questi valori, la rinumerazione parte dalla linea 10 con incrementi di 10.

Questo comando rinumererà anche i GOTO, GOSUB, ON-GOTO, ON-GOSUB, IF-THEN, etc. Un cenno va fatto al GOTO (con uno spazio tra GO e TO) che non viene rinumerato. Usate FIND per vedere se esiste qualche GO TO nel vostro programma, e trasformatelo in GOTO prima di usare RENUM. Esempio: RENUM100,20 rinumererà il programma a partire da 100 con incrementi di 20.

RESAVE - Comando per disk drive

Sintassi: **RESAVE** "nome file"

E' un 'Save con replace' che risolve tutti i problemi che a volte si incontrano con il solito comando SAVE"@:nome file". Quest'ultimo, infatti, prima salva il nuovo file e poi cancella il precedente, in modo che - se non vi è spazio sufficiente sul disco - si possono avere dei problemi. Il classico 'Save con replace' è a volte inaffidabile e può provocare la perdita di entrambi i file e, anche se alcuni esperti non sono d'accordo su questa affermazione, è quindi da evitare il più possibile.

sibile.

RESAVE, al contrario, cancella il vecchio file prima di salvare il nuovo, ed è quindi scevro da tutti i rischi insiti nel SAVE"@:nome file": può essere usato con la massima tranquillità.

RESET - Comando di MetaBasic Plus (Vedi anche QUIT)

Sintassi: **RESET**

Disabilita MetaBasic Plus. A differenza di QUIT, provoca una de-allocazione totale della memoria occupata da MetaBasic Plus: esso potrà essere riattivato con la SYS 36864 soltanto se nessun file è stato caricato e non sono state create nuove stringhe.

SCRATCH - Comando per disk drive

Sintassi: **SCRATCH** "nome file"

SCRATCH esegue la stessa funzione del comando OPEN15,8,15:PRINT#15,"S:nome file".CLOSE15, con una singola istruzione. Potete usare l'asterisco per cancellare più di un programma: SCRATCH"A*" cancellerà per esempio tutti i file su disco il cui nome inizia per A.

SCREEN - Comando per la stampante

Sintassi: **SCREEN**

Invia alla stampante il contenuto dello schermo. Il comando SCREEN funziona soltanto in modo testo e non considera schermate contenenti caratteri ridisegnati o pagine grafiche in alta risoluzione.

Il comando SCREEN equivale all'istruzione OPEN1,4,255. Se la vostra stampante richiede per questa istruzione un indirizzo secondario pari a 7, inserite il valore 7 con una POKE nella locazione 35369 e risalvate MetaBasic Plus con il comando BSAVE. Se desiderate modificare il numero di periferica utilizzato da SCREEN, inserite con una POKE il nuovo valore nella locazione 35367 e risalvate MetaBasic Plus tramite BSAVE.

SEND - Comando per disk drive

Sintassi: **SEND** "comando"

E' un modo molto conveniente per inviare comandi al drive attraverso il canale 15. SEND"I" inizializza il drive, SEND"V" esegue una validate, SEND"R:novo=vecchio" cambia il nome ad un file, e così via.

SPEED - Programmazione BASIC

Sintassi: **SPEED** numero

SPEED seguito da un numero tra 0 e 255, modifica la velocità di stampa. Più alto è il numero, più bassa è la velocità. Provate a digitare SPEED255 e quindi listate un programma. Potete ritornare in condizioni normali con SPEED0.

Se questo comando non funziona, usate prima INT per ripristinare correttamente gli interrupt. SPEED è molto utile anche nell'uso del comando TRACE (vedi più avanti).

START - Comando per disk drive

Sintassi: **START "nome file"**

Se vi siete dimenticati l'indirizzo iniziale di un programma in linguaggio macchina che avete su disco (cioè il numero da dare - nella maggior parte dei casi - nella SYS di attivazione), questo comando lo visualizza.

SUPERPEEK - Programmazione BASIC o L/M

Sintassi: **SUPERPEEK inizio, numero**

Crea una finestra nella parte alta dello schermo, in cui viene visualizzato il contenuto di un'area di memoria tramite numeri decimali. La visualizzazione inizia con l'indirizzo 'inizio' e comprende il numero di byte specificato in 'numero'. La pressione di RUN/STOP provoca la visualizzazione dei codici caratteri equivalenti ai numeri della finestra. SUPERPEEK viene disabilitato dal comando INT.

TERMINAL - Comando per modem

Sintassi: **TERMINAL**

Se disponete di un modem Commodore e lo avete collegato al C64, il comando **TERMINAL** trasforma il vostro computer in un 'terminale stupido' a 300 baud che potete usare per comunicare con servizi di informazione che usino l'ASCII standard. Non potete, con questo comando, modificare alcuno dei parametri di default (come il full duplex), né potete caricare in memoria o salvare via modem testi.

TRACE - Programmazione BASIC (vedi anche TROFF)

Sintassi: **TRACE seguito da RUN**

Se state cercando qualche errore in un programma BASIC, TRACE è molto utile.

Mano mano che una linea viene eseguita, il suo numero viene visualizzato sullo schermo. Usate SHIFT o CTRL per fermare temporaneamente l'esecuzione del programma. SPEED controlla la velocità di esecuzione, e TROFF disabilita il TRACE.

TRAP - Programmazione BASIC (Vedi anche UNTRAP)

Sintassi: **TRAP**

Abilita la diagnostica degli errori di MetaBasic Plus: quando si verifica un errore durante l'esecuzione di un programma BASIC, viene listata la linea che ha provocato l'errore con il messaggio "ERROR" in campo inverso, situato in corrispondenza dell'istruzione che ha provocato l'errore. TRAP funziona soltanto durante l'esecuzione di un programma e non in modo diretto.

TROFF - Programmazione BASIC

Sintassi: **TROFF**

Disabilita la funzione TRACE.

UNNEW - Programmazione BASIC

Sintassi: **UNNEW**

Se accidentalmente digitate NEW e perdete un programma, potrete recuperarlo digitando UNNEW.

UNTRAP Programmazione BASIC (Vedi anche TRAP)

Sintassi: **UNTRAP**

Disabilita la diagnostica degli errori di MetaBasic Plus che è stata attivata con l'istruzione TRAP.

VCHANGE - Programmazione BASIC (vedi anche CHANGE, FIND)

Sintassi: **VCHANGE@VECCIO@NO-VO@,(inizio,fine)**

oppure: **VCHANGE@"VECCIO"@NUOVO@,(inizio,fine)**

VCHANGE (Verify CHANGE) funziona come CHANGE, con l'eccezione che vi viene data possibilità di decidere, caso per caso, se effettuare o meno il cambiamento.

Ogni linea contenente la vecchia stringa viene visualizzata, con ogni occorrenza della stringa evidenziata da un cerchietto pieno. Se premete Y, il cambiamento viene effettuato. Se premete N, si passa alla linea successiva.

@ - Programmazione L/M

Sintassi: **@ind.iniziale,numero,numero,...**

Funziona come una POKE, con la differenza che permette di inserire una serie di numeri in locazioni di memoria consecutive. Per esempio, se volete portare i colori di bordo e sfondo in bianco, potrete usare @53280,1,1. Il primo 1 viene inserito in 53280, mentre il secondo in 53281. Se aggiungete altri numeri, separati da virgole, verranno immessi nelle locazioni 53282, 53283, e così via...

Questo comando può inoltre essere utilizzato insieme a MEMORY. Per prima cosa, fate l'equivalente di una PEEK in una serie di locazioni di memoria con MEMORY.

Quindi modificate il contenuto di queste locazioni ponendo una @ all'inizio di ogni linea di byte che volete modificare. Andate con il cursore sui numeri che volete modificare, modificateli, e premete RETURN.

Note per il caricamento e la digitazione di MetaBasic Plus

Se si dispone della cassetta allegata alla rivista, è sufficiente posizionarsi con il nastro in corrispondenza dell'inizio del programma e digitare:

LOAD"" ,1,1 (RETURN)

(attenzione: non inserire spazi tra gli apici)

Il programma andrà quindi mandato in esecuzione digitando:

NEW

SYS36864

(RETURN)

(RETURN)

Ricordiamo che per conoscere il punto esatto di inizio dei vari programmi su cassetta è sufficiente inserire quest'ultima completamente riavvolta nel registratore, azzerare il contanastro e digitare:

LOAD""

(RETURN)

Dal momento che sicuramente non esiste sulla cassetta alcun programma di nome "", il computer passerà in rassegna tutti i programmi incontrati visualizzandone il nome con il messaggio:

FOUND XXXX

Sarà quindi utile annotarsi su un foglio i nomi dei programmi e la relativa posizione del contanastro (sottrarre tre o quattro unità dal valore corrispondente all'apparizione del messaggio FOUND...). Gli abbonati alla sola rivista dovranno invece usare il programma MLX (vedere nelle pagine verdi) per digitare il listato in linguaggio macchina. La procedura da seguire, in questo caso, è:

1. Accendere il computer e digitare:

POKE644,137:SYS58260 (RETURN)

Caricare in memoria MLX (che sarà stato precedentemente digitato e salvato), e rispondere alle sue richieste circa gli indirizzi iniziale e finale con:

indirizzo iniziale: 35145

indirizzo finale: 40959

Dopo avere digitato e salvato il programma tramite MLX, si potranno seguire per il caricamento le istruzioni date più sopra per i possessori di cassetta.

Note tecniche

La prima cosa eseguita da MetaBasic Plus è lo spostamento in basso del puntatore di fine memoria, per proteggere se stesso dalle variabili. Dopo la SYS di attivazione, può sembrare che non accada nulla di particolare: in realtà, MetaBasic Plus sta 'lavorando' in sottofondo, e vi trovate ad avere i nuovi 32 comandi prima descritti.

Digitate sempre NEW dopo aver caricato MetaBasic Plus. Una caratteristica che funziona in modo automatico è la pausa nel LIST. Quando state listando un programma, tenete premuti CTRL, SHIFT o il tasto Commodore per bloccare temporaneamente il listing.

Se desiderate interrompere l'esecuzione dei vari comandi come per esempio RENUM, usate solo RUN/STOP, non RUN/STOP+RESTORE!

I vari comandi funzionano solo in modo

diretto: non è possibile inserirli nei vostri programmi. Al contrario dei comandi BASIC ordinari, non sono ammesse ab-

breviazioni.

Infine, assicuratevi di caricare MetaBa-

sic Plus prima di qualunque altro programma, sia esso in BASIC od in L/M.

MetaBasic Plus

35145	:255,191,255,063,255,000,068	35463	:210,255,162,000,076,057,127
35151	:083,067,082,069,069,206,143	35469	:138,169,013,032,210,255,190
35157	:013,138,000,083,085,080,228	35475	:032,231,255,076,055,158,186
35163	:069,082,080,069,069,203,151	35481	:234,169,147,032,128,140,235
35169	:154,138,000,084,082,065,108	35487	:173,152,000,032,146,145,039
35175	:080,160,209,139,000,085,008	35493	:140,017,139,140,160,138,131
35181	:078,084,082,065,208,115,229	35499	:141,022,139,141,161,138,145
35187	:140,000,077,065,075,069,029	35505	:032,146,145,192,050,144,118
35193	:068,065,084,193,159,140,062	35511	:002,160,050,140,027,139,189
35199	:000,067,085,082,083,079,011	35517	:140,253,138,173,020,003,148
35205	:082,070,082,069,197,213,078	35523	:201,037,208,010,173,021,077
35211	:141,000,067,085,082,083,085	35529	:003,201,139,208,003,076,063
35217	:079,082,080,082,073,078,107	35535	:221,138,173,020,003,141,135
35223	:212,068,142,000,072,069,202	35541	:205,139,173,021,003,141,127
35229	:076,208,089,142,000,069,229	35547	:206,139,173,161,138,174,186
35235	:078,196,194,142,000,070,075	35553	:160,138,134,199,032,205,069
35241	:082,069,069,077,069,205,228	35559	:189,169,013,032,210,255,075
35247	:205,142,000,080,079,073,242	35565	:024,173,160,138,105,007,076
35253	:078,084,069,210,221,142,217	35571	:141,160,138,144,003,238,043
35259	:000,082,069,083,069,212,190	35577	:161,138,056,169,255,233,237
35265	:003,143,000,000,175,175,177	35583	:007,141,253,138,176,216,162
35271	:175,160,255,162,255,232,158	35589	:120,169,037,141,020,003,239
35277	:189,000,002,201,032,240,101	35595	:169,139,141,021,003,169,141
35283	:248,141,231,137,200,185,073	35601	:145,141,063,139,169,000,162
35289	:078,137,208,250,200,185,251	35607	:141,064,139,169,006,141,171
35295	:078,137,208,003,076,235,192	35613	:056,139,088,162,128,076,166
35301	:145,201,066,208,237,232,038	35619	:055,158,162,004,142,174,218
35307	:200,185,078,137,048,008,123	35625	:139,162,255,160,005,169,163
35313	:221,000,002,240,244,076,000	35631	:000,141,168,139,141,173,041
35319	:202,137,200,185,078,137,162	35637	:139,232,224,006,144,003,033
35325	:141,010,138,200,185,078,237	35643	:076,204,139,189,145,000,044
35331	:137,141,011,138,134,122,174	35649	:141,119,139,141,078,139,054
35337	:076,089,142,234,032,231,045	35655	:165,203,201,063,208,036,179
35343	:255,160,000,140,058,138,254	35661	:169,000,201,032,176,003,146
35349	:160,004,140,059,138,169,179	35667	:076,106,139,201,096,144,077
35355	:232,141,005,137,169,003,202	35673	:003,076,106,139,201,064,166
35361	:141,006,137,169,001,162,137	35679	:176,003,076,108,139,056,141
35367	:004,160,255,032,186,255,163	35685	:233,064,076,108,139,169,122
35373	:032,192,255,162,001,032,207	35691	:032,009,128,076,115,139,094
35379	:201,255,160,000,162,000,061	35697	:169,032,032,172,139,169,058
35385	:173,231,007,041,127,201,069	35703	:000,201,100,144,008,233,037
35391	:032,176,005,105,064,076,009	35709	:100,238,168,139,076,120,198
35397	:090,138,201,064,176,003,229	35715	:139,032,164,139,201,010,048
35403	:076,090,138,201,096,176,084	35721	:144,008,233,010,238,168,170
35409	:005,105,032,076,090,138,015	35727	:139,076,135,139,032,164,060
35415	:024,105,064,032,210,255,009	35733	:139,141,168,139,032,164,164
35421	:206,005,137,173,005,137,244	35739	:139,169,032,032,172,139,070
35427	:201,255,208,003,206,006,210	35745	:076,054,139,141,202,139,144
35433	:137,173,005,137,208,005,002	35751	:169,000,024,105,048,153,154
35439	:173,006,137,240,026,238,163	35757	:000,004,200,192,040,208,049
35445	:058,138,173,058,138,208,122	35763	:016,160,005,024,173,173,218
35451	:003,238,059,138,232,224,249	35769	:139,105,040,141,173,139,154
35457	:040,144,007,169,013,032,022	35775	:144,003,238,174,139,169,034
		35781	:000,141,168,139,169,000,046

35787	:096,076,065,147,234,234,031	36141	:032,032,187,141,173,050,148
35793	:169,227,141,000,003,169,150	36147	:141,201,100,144,008,233,110
35799	:139,141,001,003,169,000,156	36153	:100,238,183,141,076,052,079
35805	:141,237,139,076,055,158,003	36159	:141,032,179,141,201,010,255
35811	:165,058,201,255,208,003,093	36165	:144,008,233,010,238,183,117
35817	:076,139,227,169,000,208,028	36171	:141,076,067,141,032,179,199
35823	:075,224,128,240,245,142,013	36177	:141,141,183,141,032,179,130
35829	:092,140,165,057,133,020,084	36183	:141,206,092,141,169,183,251
35835	:165,058,133,021,032,019,167	36189	:201,255,208,003,206,106,048
35841	:166,165,095,133,155,165,112	36195	:141,173,092,141,208,007,093
35847	:096,133,156,160,003,140,183	36201	:169,001,208,003,076,138,188
35853	:060,140,056,165,122,229,017	36207	:141,238,050,141,173,050,136
35859	:155,141,063,140,169,013,188	36213	:141,208,003,238,051,141,131
35865	:141,237,139,032,210,255,015	36219	:224,075,144,003,076,232,109
35871	:032,210,255,165,058,133,116	36225	:140,169,044,032,187,141,074
35877	:098,165,057,133,099,162,239	36231	:076,049,141,169,000,032,090
35883	:144,056,032,073,188,032,056	36237	:187,141,173,188,141,160,107
35889	:223,189,032,030,171,169,095	36243	:000,145,251,173,189,141,022
35895	:032,032,210,255,160,099,075	36249	:200,145,251,169,000,032,182
35901	:200,192,099,208,003,032,027	36255	:187,141,169,000,032,187,107
35907	:096,140,177,155,141,082,090	36261	:141,173,188,141,133,045,218
35913	:140,240,010,140,060,140,035	36267	:173,189,141,133,046,076,161
35919	:160,255,169,099,076,026,096	36273	:055,158,141,208,141,169,025
35925	:167,169,000,141,237,139,170	36279	:000,024,105,048,141,169,158
35931	:162,099,076,139,227,169,195	36285	:027,238,188,141,172,188,119
35937	:107,140,109,140,160,163,148	36291	:141,208,003,238,189,141,091
35943	:132,199,032,030,171,160,059	36297	:160,000,140,183,141,232,033
35949	:099,169,000,133,199,096,037	36303	:169,009,096,175,175,173,236
35955	:169,139,141,000,003,169,224	36309	:120,173,143,002,201,250,078
35961	:227,141,001,003,076,055,112	36315	:208,003,076,082,142,141,103
35967	:158,032,210,255,173,134,065	36321	:011,142,173,144,002,141,070
35973	:002,160,250,153,000,216,146	36327	:012,142,169,250,141,143,064
35979	:153,250,216,153,244,217,092	36333	:002,169,141,141,144,002,068
35985	:153,238,218,136,208,241,059	36339	:141,007,142,088,076,082,011
35991	:096,141,001,003,108,002,246	36345	:142,165,203,240,060,201,236
35997	:003,002,032,146,145,132,105	36351	:002,240,011,201,007,240,188
36003	:251,140,050,141,133,252,106	36357	:007,169,044,240,029,076,058
36009	:141,051,141,032,146,145,057	36363	:072,235,165,212,208,007,142
36015	:132,253,133,254,056,165,144	36369	:165,216,208,030,076,010,210
36021	:253,229,251,141,092,141,008	36375	:142,165,212,141,045,142,102
36027	:165,254,229,252,141,106,054	36381	:169,000,141,007,142,133,109
36033	:141,173,106,141,201,017,204	36387	:212,076,010,142,169,044,176
36039	:144,005,162,014,108,000,120	36393	:141,007,142,169,000,133,121
36045	:003,169,000,141,188,141,079	36399	:212,076,010,142,169,000,144
36051	:141,022,141,141,183,141,212	36405	:133,216,076,010,142,169,031
36057	:169,008,141,189,141,169,010	36411	:000,141,045,142,133,212,220
36063	:076,141,249,140,169,008,238	36417	:076,010,142,120,173,011,085
36069	:141,254,140,162,000,138,040	36423	:142,141,143,002,173,012,172
36075	:032,187,141,173,188,141,073	36429	:142,141,144,002,088,162,244
36081	:133,251,173,189,141,133,237	36435	:128,108,000,003,002,002,070
36087	:252,169,180,032,187,141,184	36441	:169,147,032,128,140,160,097
36093	:169,024,032,187,141,160,198	36447	:255,162,000,142,146,142,174
36099	:000,024,173,249,140,105,182	36453	:142,189,142,169,004,141,120
36105	:075,141,249,140,173,254,017	36459	:190,142,200,185,078,137,015
36111	:140,105,000,141,254,140,027	36465	:208,250,200,185,078,137,147
36117	:169,058,032,187,141,169,009	36471	:208,011,169,005,133,214,091
36123	:000,032,187,141,238,022,135	36477	:169,000,133,251,076,251,237
36129	:141,169,131,032,187,141,066	36483	:157,016,045,201,160,208,150
36135	:169,032,032,187,141,169,001	36489	:002,169,096,041,127,032,092

36495	:185,142,169,000,208,008,087	36849	:000,000,000,000,000,000,241
36501	:162,020,142,146,142,076,069	36855	:000,000,000,000,000,000,247
36507	:109,142,162,000,142,146,088	36861	:000,000,000,076,194,145,156
36513	:142,024,173,189,142,105,168	36867	:069,001,008,000,000,255,080
36519	:040,144,003,238,190,142,156	36873	:255,000,001,000,000,064,073
36525	:141,189,142,076,109,142,204	36879	:000,000,000,199,008,000,222
36531	:032,185,142,076,115,142,103	36885	:012,012,066,006,000,000,117
36537	:056,233,064,157,240,004,171	36891	:004,004,010,000,000,000,045
36543	:232,096,175,165,046,166,047	36897	:000,000,000,000,169,001,203
36549	:045,133,098,134,099,076,014	36903	:044,169,000,141,036,144,061
36555	:242,142,056,165,051,229,064	36909	:032,146,145,169,069,141,235
36561	:049,133,099,165,052,229,168	36915	:040,003,169,145,141,041,078
36567	:050,133,098,076,242,142,188	36921	:003,173,029,144,133,183,210
36573	:032,146,145,132,251,133,036	36927	:134,187,132,188,160,002,098
36579	:252,160,000,177,251,133,176	36933	:169,000,153,086,002,136,103
36585	:099,200,177,251,133,098,167	36939	:016,250,169,008,133,186,069
36591	:076,242,142,162,144,056,037	36945	:169,096,133,185,032,108,036
36597	:032,073,188,032,223,189,214	36951	:144,144,021,169,002,032,087
36603	:032,030,171,162,128,108,114	36957	:195,255,169,015,032,195,186
36609	:000,003,169,000,133,055,105	36963	:255,032,231,255,162,005,015
36615	:169,160,133,056,032,092,137	36969	:108,000,003,032,213,243,192
36621	:166,032,083,228,000,000,010	36975	:165,186,032,180,255,165,070
36627	:255,255,255,255,255,255,013	36981	:185,032,150,255,169,152,036
36633	:255,255,255,255,255,255,019	36987	:141,000,003,169,144,141,209
36639	:255,000,000,000,000,000,030	36993	:001,003,169,044,141,002,233
36645	:000,000,000,000,000,000,037	36999	:003,169,145,141,003,003,087
36651	:000,000,000,000,000,000,043	37005	:032,165,255,032,080,145,082
36657	:000,000,000,000,000,000,049	37011	:176,072,032,165,255,032,111
36663	:000,000,000,000,000,000,055	37017	:165,255,141,023,144,032,145
36669	:000,000,000,255,255,255,058	37023	:165,255,013,023,144,240,231
36675	:255,255,255,255,255,255,061	37029	:055,032,165,255,133,020,057
36681	:255,255,255,255,255,255,067	37035	:141,023,144,032,165,255,163
36687	:255,255,255,255,255,255,073	37041	:133,021,141,024,144,173,045
36693	:255,255,255,255,255,255,079	37047	:036,144,208,004,160,001,224
36699	:255,255,255,255,255,000,086	37053	:208,002,160,255,200,032,022
36705	:000,000,000,000,000,000,097	37059	:165,255,153,000,002,208,210
36711	:000,000,000,000,000,000,103	37065	:247,032,060,145,032,137,086
36717	:000,000,000,000,000,000,109	37071	:145,152,024,105,005,168,038
36723	:000,000,000,000,000,000,115	37077	:173,036,144,240,045,076,159
36729	:000,000,000,000,000,000,121	37083	:162,164,032,171,255,032,011
36735	:000,223,255,255,255,255,090	37089	:066,246,032,231,255,169,200
36741	:255,255,255,255,255,255,127	37095	:131,141,002,003,169,164,073
36747	:255,255,255,255,255,255,133	37101	:141,003,003,169,139,141,065
36753	:255,255,255,255,255,255,139	37107	:000,003,169,227,141,001,016
36759	:255,255,255,255,255,255,145	37113	:003,169,237,141,040,003,074
36765	:255,255,255,000,000,000,154	37119	:169,246,141,041,003,076,163
36771	:000,000,000,000,000,000,163	37125	:055,158,032,033,159,174,104
36777	:000,000,000,000,000,000,169	37131	:023,144,173,024,144,032,039
36783	:000,000,000,000,000,000,175	37137	:205,189,169,000,133,095,040
36789	:000,000,000,000,000,000,181	37143	:169,002,133,096,169,000,080
36795	:000,000,000,000,000,255,186	37149	:133,073,169,086,141,000,119
36801	:255,255,255,255,255,255,187	37155	:002,169,002,141,001,002,096
36807	:255,255,255,255,255,255,193	37161	:076,237,166,173,023,144,092
36813	:255,255,255,255,255,255,199	37167	:133,020,173,024,144,133,162
36819	:255,255,255,255,255,255,205	37173	:021,032,019,166,076,201,056
36825	:255,255,255,255,255,255,211	37179	:166,072,165,203,201,063,161
36831	:255,000,000,000,000,000,222	37185	:240,154,104,096,072,165,128
36837	:000,000,000,000,000,000,229	37191	:203,201,063,240,145,104,003
36843	:000,000,000,000,000,000,235	37197	:076,237,246,141,018,144,171

37203	:024,165,144,240,045,032,221	37557	:082,071,197,037,144,077,021
37209	:033,159,169,015,162,008,123	37563	:079,078,073,084,079,210,022
37215	:168,032,186,255,169,000,137	37569	:004,150,078,085,077,066,141
37221	:032,189,255,032,192,255,032	37575	:069,210,166,150,081,085,192
37227	:162,015,032,198,255,032,033	37581	:073,212,001,150,082,069,024
37233	:207,255,072,032,210,255,120	37587	:065,196,103,148,082,069,106
37239	:104,201,013,208,244,169,034	37593	:078,085,205,000,153,082,052
37245	:015,032,195,255,032,204,090	37599	:069,083,065,086,197,014,225
37251	:255,056,173,018,144,096,105	37605	:150,083,067,082,065,084,248
37257	:160,255,200,185,000,002,171	37611	:067,200,005,150,083,069,041
37263	:208,250,096,032,121,000,082	37617	:078,196,025,150,083,080,085
37269	:201,000,240,039,032,115,008	37623	:069,069,196,229,157,083,026
37275	:000,144,008,201,034,240,014	37629	:084,065,082,212,106,148,182
37281	:004,056,076,181,145,032,143	37635	:084,069,082,077,073,078,210
37287	:158,173,165,013,208,009,125	37641	:065,204,064,158,084,082,154
37293	:032,155,188,165,100,164,209	37647	:065,067,197,174,152,084,242
37299	:101,024,096,032,130,183,233	37653	:082,079,070,198,243,152,077
37305	:141,029,144,166,034,164,095	37659	:085,078,078,069,215,155,195
37311	:035,056,096,120,169,200,099	37665	:150,086,067,072,065,078,039
37317	:141,004,003,169,137,141,024	37671	:071,197,188,155,064,160,106
37323	:005,003,169,065,141,020,094	37677	:208,147,160,001,185,061,039
37329	:003,169,147,141,021,003,181	37683	:147,145,122,200,192,004,093
37335	:032,062,148,088,165,056,254	37689	:208,246,076,040,144,034,037
37341	:201,137,144,007,169,137,248	37695	:036,000,165,203,205,014,174
37347	:133,056,032,092,166,076,014	37701	:144,240,074,141,014,144,058
37353	:055,158,162,255,232,189,004	37707	:201,003,208,004,169,007,155
37359	:000,002,201,032,240,248,194	37713	:208,008,201,004,144,059,193
37365	:160,255,141,023,144,200,144	37719	:201,007,176,055,056,233,047
37371	:192,224,176,079,217,081,196	37725	:004,010,010,010,010,010,147
37377	:146,208,024,140,024,144,175	37731	:174,141,002,240,003,024,171
37383	:142,025,144,232,200,185,167	37737	:105,016,168,162,000,185,229
37389	:081,146,048,029,221,000,026	37743	:097,159,201,095,208,002,105
37395	:002,240,244,172,024,144,077	37749	:169,013,201,039,208,008,243
37401	:174,025,144,200,192,224,216	37755	:165,212,073,001,133,212,151
37407	:176,045,185,081,146,016,168	37761	:169,034,201,000,240,009,014
37413	:246,173,023,144,200,200,255	37767	:157,119,002,232,200,224,045
37419	:076,250,145,142,027,144,059	37773	:010,208,222,134,198,076,221
37425	:134,122,232,189,000,002,216	37779	:049,234,032,146,145,136,121
37431	:240,004,201,065,176,017,246	37785	:152,048,047,201,008,176,017
37437	:200,185,081,146,141,076,122	37791	:043,010,010,010,010,024,010
37443	:146,200,185,081,146,141,198	37797	:105,097,133,251,169,159,055
37449	:077,146,076,039,159,076,134	37803	:105,000,133,252,032,146,071
37455	:124,165,065,085,084,207,041	37809	:145,144,023,173,029,144,067
37461	:021,152,066,083,065,086,046	37815	:201,011,176,016,168,169,156
37467	:197,039,159,067,065,212,062	37821	:000,145,251,136,177,034,164
37473	:047,147,067,072,065,078,061	37827	:145,251,136,016,249,076,044
37479	:071,197,212,155,068,069,107	37833	:055,158,162,011,108,000,183
37485	:070,065,085,076,212,242,091	37839	:003,169,255,133,251,032,026
37491	:147,068,069,076,069,084,116	37845	:166,145,176,242,140,234,036
37497	:197,182,148,068,076,073,097	37851	:147,141,235,147,032,146,043
37503	:083,212,040,144,068,085,247	37857	:145,176,011,152,230,251,166
37509	:077,208,087,151,069,082,039	37863	:164,251,153,000,000,076,107
37515	:210,056,148,070,073,078,006	37869	:223,147,108,002,003,120,072
37521	:196,200,155,072,069,076,145	37875	:169,095,141,024,003,169,076
37527	:208,238,157,073,078,212,093	37881	:157,141,025,003,088,169,064
37533	:194,145,075,069,217,149,238	37887	:003,133,251,032,146,145,197
37539	:147,076,076,073,083,212,062	37893	:176,010,152,166,251,157,149
37545	:144,157,077,069,077,079,004	37899	:019,144,198,251,016,241,112
37551	:082,217,057,149,077,069,058	37905	:169,040,141,050,003,169,077

37911	:148,141,051,003,169,048,071	38265	:024,105,008,133,211,160,250
37917	:141,048,003,169,148,141,167	38271	:000,177,251,032,210,149,178
37923	:049,003,076,055,158,174,038	38277	:169,058,032,210,255,230,063
37929	:019,144,134,186,076,237,069	38283	:251,208,002,230,252,076,134
37935	:245,174,019,144,134,186,181	38289	:077,149,169,058,032,210,072
37941	:076,165,244,032,088,145,035	38295	:255,169,018,032,210,255,066
37947	:076,055,158,169,073,141,219	38301	:198,252,160,250,177,251,165
37953	:038,003,169,148,141,039,091	38307	:201,032,144,007,201,127,107
37959	:003,096,072,138,072,152,092	38313	:176,003,076,176,149,169,150
37965	:072,173,141,002,208,251,156	38319	:046,032,210,255,200,208,102
37971	:174,012,144,232,160,001,038	38325	:235,230,252,032,033,159,098
37977	:136,208,253,202,208,250,066	38331	:169,032,032,210,255,165,026
37983	:104,168,104,170,104,076,053	38337	:252,166,251,032,205,189,008
37989	:202,241,169,001,044,169,159	38343	:169,000,141,028,144,169,082
37995	:000,141,011,144,032,146,069	38349	:062,032,210,255,096,168,004
38001	:145,032,189,255,169,002,137	38355	:032,162,179,032,221,189,002
38007	:162,008,160,002,032,186,157	38361	:160,255,200,185,000,001,250
38013	:255,032,192,255,144,003,238	38367	:208,250,140,023,144,136,100
38019	:076,090,144,162,002,032,125	38373	:136,136,185,000,001,201,120
38025	:198,255,032,165,255,032,050	38379	:049,144,007,201,058,176,102
38031	:080,145,176,024,174,011,241	38385	:003,076,247,149,169,048,165
38037	:144,208,010,170,032,165,110	38391	:032,210,255,200,204,023,147
38043	:255,032,205,189,076,171,059	38397	:144,208,231,096,032,083,023
38049	:148,032,210,255,032,060,130	38403	:228,000,169,000,141,017,046
38055	:145,076,139,148,169,002,078	38409	:144,133,180,240,020,169,127
38061	:032,195,255,032,204,255,122	38415	:000,141,017,144,169,001,231
38067	:076,055,158,032,146,145,023	38421	:133,180,208,009,169,001,209
38073	:176,115,132,253,133,254,224	38427	:141,017,144,169,000,133,119
38079	:032,049,149,160,002,177,248	38433	:180,032,146,145,172,029,225
38085	:095,141,023,144,200,177,209	38439	:144,169,000,153,003,002,254
38091	:095,141,024,144,165,095,099	38445	:136,177,034,153,003,002,038
38097	:133,251,165,096,133,252,215	38451	:136,016,248,173,017,144,017
38103	:032,146,145,176,082,200,228	38457	:208,011,160,002,185,084,195
38109	:208,003,024,105,001,032,082	38463	:151,153,000,002,136,016,009
38115	:049,149,056,165,101,229,208	38469	:247,169,015,162,008,160,062
38121	:253,165,100,229,254,144,098	38475	:015,032,186,255,032,137,220
38127	:062,160,000,140,000,002,091	38481	:145,152,174,017,144,240,185
38133	:165,095,145,251,200,165,242	38487	:009,056,233,003,162,003,041
38139	:096,145,251,173,023,144,059	38493	:160,002,208,004,162,000,117
38145	:133,020,173,024,144,133,116	38499	:160,002,032,189,255,032,001
38151	:021,160,005,173,002,003,115	38505	:192,255,144,003,076,090,097
38157	:133,251,173,003,003,133,197	38511	:144,169,015,032,195,255,153
38163	:252,169,033,141,002,003,107	38517	:165,180,240,028,169,003,134
38169	:169,149,141,003,003,076,054	38523	:162,008,160,003,032,186,162
38175	:162,164,165,251,141,002,148	38529	:255,173,029,144,162,003,127
38181	:003,165,252,141,003,003,092	38535	:160,002,032,189,255,169,174
38187	:076,049,158,076,203,147,240	38541	:043,166,045,164,046,032,125
38193	:132,020,133,021,032,019,150	38547	:216,255,032,080,145,076,183
38199	:166,096,032,146,145,132,004	38553	:055,158,160,001,152,145,056
38205	:251,133,252,032,146,145,252	38559	:043,032,051,165,076,049,063
38211	:132,253,133,254,032,184,031	38565	:158,032,146,145,176,050,104
38217	:149,076,116,149,032,060,143	38571	:132,251,133,252,072,169,156
38223	:145,056,165,253,229,251,154	38577	:036,032,210,255,104,032,078
38229	:165,254,229,252,176,010,147	38583	:192,150,152,032,192,150,027
38235	:173,028,144,201,001,208,078	38589	:076,240,150,072,074,074,107
38241	:003,076,055,158,238,028,143	38595	:074,074,032,207,150,104,068
38247	:144,173,028,144,201,006,031	38601	:041,015,032,207,150,096,230
38253	:208,005,169,001,032,147,159	38607	:201,010,144,003,024,105,182
38259	:149,173,028,144,010,010,117	38613	:007,024,105,048,032,210,127

38619	:255,096,169,000,133,251,099	38973	:133,020,173,035,144,133,187
38625	:133,252,168,200,177,122,253	38979	:021,120,169,093,141,002,101
38631	:208,071,165,252,166,251,064	38985	:003,169,152,141,003,003,032
38637	:032,205,189,032,033,159,119	38991	:169,141,141,020,003,169,210
38643	:169,000,166,251,032,205,042	38997	:152,141,021,003,088,076,054
38649	:189,169,032,032,210,255,112	39003	:055,158,165,020,024,109,110
38655	:169,000,166,252,032,205,055	39009	:032,144,133,099,165,021,179
38661	:189,032,033,159,169,037,112	39015	:109,033,144,133,098,162,014
38667	:032,210,255,169,000,141,050	39021	:144,056,032,073,188,032,122
38673	:028,144,162,016,169,048,072	39027	:221,189,162,000,189,001,109
38679	:006,251,038,252,144,005,207	39033	:001,240,006,157,119,002,134
38685	:169,049,238,028,144,172,061	39039	:232,208,245,169,032,157,146
38691	:028,144,240,003,032,210,180	39045	:119,002,232,134,198,076,126
38697	:255,202,208,232,076,055,045	39051	:131,164,032,225,255,208,130
38703	:158,162,004,006,251,038,154	39057	:025,120,169,131,141,002,221
38709	:252,202,208,249,032,074,046	39063	:003,169,164,141,003,003,122
38715	:151,024,101,251,133,251,202	39069	:169,065,141,020,003,169,212
38721	:165,252,105,000,133,252,204	39075	:147,141,021,003,032,033,028
38727	:076,228,150,056,233,048,094	39081	:159,088,076,065,147,169,105
38733	:201,010,144,002,233,007,162	39087	:195,141,008,003,169,152,075
38739	:096,083,048,058,032,088,232	39093	:141,009,003,169,000,141,132
38745	:159,234,165,046,133,252,054	39099	:030,144,141,031,144,076,241
38751	:076,116,151,169,058,032,185	39105	:055,158,165,058,201,255,061
38757	:210,255,165,251,024,105,087	39111	:240,039,173,030,144,197,254
38763	:007,133,251,165,252,105,252	39117	:057,208,009,173,031,144,059
38769	:000,133,252,160,000,056,202	39123	:197,058,208,002,240,023,171
38775	:165,251,229,047,165,252,204	39129	:169,091,032,210,255,166,116
38781	:229,048,144,005,169,000,208	39135	:057,142,030,144,165,058,051
38787	:076,049,168,169,000,141,222	39141	:141,031,144,032,201,189,199
38793	:028,144,177,251,016,004,245	39147	:169,093,032,210,255,076,046
38799	:056,046,028,144,041,127,073	39153	:228,167,169,228,141,008,158
38805	:032,210,255,200,177,251,250	39159	:003,169,167,141,009,003,227
38811	:016,004,056,046,028,144,193	39165	:076,055,158,032,157,155,118
38817	:041,127,240,003,032,210,046	39171	:169,010,141,006,144,141,102
38823	:255,173,028,144,208,024,231	39177	:008,144,169,000,141,011,226
38829	:032,004,152,200,152,024,225	39183	:144,141,007,144,141,009,089
38835	:101,251,072,165,252,105,101	39189	:144,032,146,145,176,017,169
38841	:000,168,104,032,162,187,070	39195	:140,006,144,141,007,144,097
38847	:032,010,152,076,098,151,198	39201	:032,146,145,176,006,140,166
38853	:201,003,208,026,169,037,073	39207	:008,144,141,009,144,160,133
38859	:032,210,255,032,004,152,120	39213	:000,177,188,141,023,144,206
38865	:200,177,251,072,200,177,006	39219	:200,177,188,013,023,144,028
38871	:251,168,104,032,145,179,070	39225	:240,058,200,200,200,177,108
38877	:032,010,152,076,098,151,228	39231	:188,240,074,201,044,208,250
38883	:201,001,208,026,169,036,100	39237	:005,173,011,144,208,081,179
38889	:032,210,255,032,004,152,150	39243	:234,234,234,234,201,137,069
38895	:200,177,251,072,200,177,036	39249	:240,073,201,141,240,069,021
38901	:251,133,034,200,177,251,011	39255	:201,167,208,019,140,023,077
38907	:133,035,104,032,036,171,250	39261	:144,200,177,188,201,032,011
38913	:076,098,151,169,061,032,076	39267	:240,249,172,023,144,201,104
38919	:210,255,096,032,221,189,242	39273	:058,144,048,176,207,169,139
38925	:072,152,170,104,032,030,061	39279	:000,141,011,144,240,200,079
38931	:171,096,032,146,145,140,237	39285	:169,001,141,028,144,169,001
38937	:034,144,141,035,144,032,043	39291	:255,133,100,133,101,032,109
38943	:146,145,140,032,144,141,011	39297	:051,165,032,198,154,032,249
38949	:033,144,173,034,144,056,109	39303	:146,155,076,049,158,200,151
38955	:237,032,144,141,034,144,007	39309	:152,024,101,188,133,188,159
38961	:173,035,144,237,033,144,047	39315	:165,189,105,000,133,189,160
38967	:141,035,144,173,034,144,214	39321	:076,044,153,140,015,144,213

39327 :200,177,188,201,032,240,173
 39333 :249,172,015,144,201,058,236
 39339 :176,004,201,048,176,003,011
 39345 :076,061,153,169,001,141,010
 39351 :011,144,200,024,152,101,047
 39357 :188,133,122,133,186,165,092
 39363 :189,105,000,133,123,133,110
 39369 :187,140,023,144,032,073,032
 39375 :155,169,000,141,028,144,076
 39381 :165,122,133,146,165,123,043
 39387 :133,147,032,051,165,032,011
 39393 :198,154,032,084,155,024,104
 39399 :101,186,133,148,165,187,127
 39405 :105,000,133,149,169,004,029
 39411 :141,029,144,032,013,154,244
 39417 :172,023,144,162,255,232,213
 39423 :189,000,001,208,003,076,220
 39429 :062,153,145,188,200,076,061
 39435 :254,153,032,033,154,176,045
 39441 :014,024,165,182,109,029,028
 39447 :144,133,182,165,183,105,167
 39453 :000,133,183,096,056,165,150
 39459 :148,229,146,141,024,144,099
 39465 :165,149,229,147,013,024,000
 39471 :144,208,001,096,144,066,194
 39477 :032,184,154,165,181,170,171
 39483 :005,180,208,001,096,024,061
 39489 :138,101,147,141,099,154,077
 39495 :165,146,141,098,154,024,031
 39501 :138,101,149,141,102,154,094
 39507 :165,148,141,101,154,232,000
 39513 :164,180,208,004,240,013,130
 39519 :160,255,185,000,000,153,080
 39525 :000,000,136,192,255,208,124
 39531 :245,206,099,154,206,102,095
 39537 :154,202,208,234,024,096,007
 39543 :032,184,154,165,146,141,173
 39549 :154,154,165,147,141,155,017
 39555 :154,165,148,141,157,154,026
 39561 :165,149,141,158,154,166,046
 39567 :181,240,032,169,000,141,138
 39573 :010,144,160,000,185,000,136
 39579 :000,153,000,000,200,204,200
 39585 :010,144,208,244,238,155,136
 39591 :154,238,158,154,224,000,071
 39597 :240,007,202,208,224,165,195
 39603 :180,208,222,056,096,056,229
 39609 :165,182,229,146,133,180,196
 39615 :165,183,229,147,133,181,205
 39621 :096,173,006,144,133,184,165
 39627 :173,007,144,133,185,173,250
 39633 :004,144,133,253,173,005,153
 39639 :144,133,254,160,000,177,059
 39645 :253,133,167,200,177,253,124
 39651 :133,168,005,167,240,075,247
 39657 :200,177,253,056,229,101,225
 39663 :141,013,144,200,177,253,143
 39669 :229,100,013,013,144,208,184
 39675 :018,173,028,144,208,012,066

39681 :169,032,032,210,255,166,097
 39687 :184,165,185,032,205,189,199
 39693 :096,176,036,173,028,144,154
 39699 :240,015,152,072,160,002,148
 39705 :165,184,145,253,200,165,113
 39711 :185,145,253,104,168,165,027
 39717 :184,024,109,008,144,133,127
 39723 :184,165,185,109,009,144,071
 39729 :076,060,155,169,255,133,129
 39735 :184,133,185,208,192,133,066
 39741 :185,165,167,133,253,165,105
 39747 :168,133,254,076,218,154,046
 39753 :032,158,173,032,155,188,043
 39759 :165,100,164,101,096,160,097
 39765 :255,200,185,000,001,208,166
 39771 :250,152,096,165,043,133,162
 39777 :251,165,044,133,252,160,078
 39783 :000,177,251,208,018,232,221
 39789 :224,003,208,015,152,056,255
 39795 :101,251,133,045,165,252,038
 39801 :105,000,133,046,096,162,151
 39807 :000,200,208,229,230,252,222
 39813 :208,225,162,044,181,146,075
 39819 :157,211,159,202,016,248,108
 39825 :096,162,044,189,211,159,238
 39831 :149,146,202,016,248,096,240
 39837 :032,135,155,165,045,024,201
 39843 :105,005,133,182,165,046,031
 39849 :105,000,133,183,165,043,030
 39855 :133,188,141,004,144,165,182
 39861 :044,133,189,141,005,144,069
 39867 :096,169,000,141,011,144,236
 39873 :169,001,141,013,144,208,101
 39879 :020,169,001,141,011,144,173
 39885 :169,000,141,013,144,240,144
 39891 :008,169,000,141,011,144,172
 39897 :141,013,144,032,124,165,068
 39903 :032,026,157,032,157,155,014
 39909 :169,255,141,008,144,141,063
 39915 :009,144,169,002,133,123,047
 39921 :032,146,145,176,026,132,130
 39927 :020,133,021,032,019,166,126
 39933 :165,095,133,188,165,096,071
 39939 :133,189,032,146,145,176,056
 39945 :006,140,008,144,141,009,201
 39951 :144,160,000,177,188,141,057
 39957 :023,144,200,177,188,013,254
 39963 :023,144,240,072,162,000,156
 39969 :200,177,188,133,020,237,220
 39975 :008,144,200,177,188,133,121
 39981 :021,237,009,144,144,002,090
 39987 :176,050,200,177,188,240,058
 39993 :029,205,011,001,208,246,245
 39999 :140,023,144,162,000,200,220
 40005 :232,189,011,001,201,064,255
 40011 :240,035,209,188,240,243,206
 40017 :172,023,144,076,053,156,193
 40023 :200,152,024,101,188,133,117
 40029 :188,165,189,105,000,133,105

40035	:189,076,016,156,032,051,107	40389	:166,169,139,141,000,003,047
40041	:165,032,146,155,076,049,216	40395	:169,227,141,001,003,162,138
40047	:158,140,024,144,032,051,148	40401	:004,032,201,255,169,013,115
40053	:165,172,023,144,136,192,181	40407	:032,210,255,032,204,255,179
40059	:004,176,001,200,132,180,048	40413	:169,002,032,195,255,076,182
40065	:177,188,141,003,144,174,188	40419	:055,158,032,146,145,140,135
40071	:013,144,240,004,169,113,050	40425	:012,144,076,055,158,169,079
40077	:145,188,032,019,166,169,092	40431	:000,133,251,169,147,032,203
40083	:159,141,000,003,169,156,007	40437	:210,255,169,004,133,214,206
40089	:141,001,003,076,201,166,229	40443	:032,033,159,160,255,200,066
40095	:169,139,141,000,003,169,012	40449	:192,219,144,003,076,055,178
40101	:227,141,001,003,173,003,201	40455	:158,185,081,146,048,006,119
40107	:144,164,180,145,188,172,140	40461	:032,210,255,076,000,158,232
40113	:024,144,173,011,144,240,145	40467	:073,128,032,210,255,165,114
40119	:003,076,054,156,174,013,147	40473	:251,073,001,133,251,208,174
40125	:144,240,022,165,197,197,130	40479	:009,169,000,133,211,032,073
40131	:203,240,252,032,060,145,103	40485	:033,159,208,004,169,020,118
40137	:165,203,201,025,240,007,018	40491	:133,211,200,200,208,207,178
40143	:201,039,208,243,076,054,004	40497	:032,094,155,032,092,166,108
40149	:156,152,024,101,188,133,199	40503	:169,000,133,198,162,128,077
40155	:146,165,189,105,000,133,189	40509	:108,000,003,169,000,133,218
40161	:147,173,023,144,024,109,077	40515	:247,169,206,133,248,169,215
40167	:029,144,133,148,032,251,200	40521	:000,133,249,169,207,133,196
40173	:156,165,148,024,101,188,251	40527	:250,032,231,255,169,001,249
40179	:133,148,032,251,156,076,015	40533	:162,032,160,159,032,189,051
40185	:002,157,165,189,105,000,099	40539	:255,169,002,170,160,003,082
40191	:133,149,096,032,013,154,064	40545	:032,186,255,032,192,255,025
40197	:172,023,144,162,255,232,225	40551	:032,197,158,032,204,255,213
40203	:189,041,001,201,064,208,203	40557	:032,228,255,201,000,240,041
40209	:003,076,054,156,145,188,127	40563	:031,201,092,240,066,168,145
40215	:200,208,240,160,000,200,007	40569	:120,162,054,134,001,185,009
40221	:185,000,002,240,056,201,201	40575	:000,162,162,055,134,001,129
40227	:064,208,246,162,255,232,178	40581	:088,072,162,002,032,201,178
40233	:200,185,000,002,240,043,199	40587	:255,104,032,210,255,032,003
40239	:201,034,240,246,157,011,168	40593	:204,255,162,002,032,198,230
40245	:001,201,064,208,238,173,170	40599	:255,032,228,255,201,000,098
40251	:011,144,208,023,162,255,094	40605	:240,203,041,127,168,120,032
40257	:232,200,185,000,002,240,156	40611	:162,054,134,001,185,000,187
40263	:018,201,034,240,246,157,199	40617	:160,162,055,134,001,088,001
40269	:041,001,201,064,208,238,062	40623	:072,032,204,255,104,032,106
40275	:142,029,144,200,132,122,084	40629	:210,255,076,106,158,032,250
40281	:096,162,011,108,000,003,213	40635	:204,255,169,002,032,195,020
40287	:120,072,138,072,152,072,209	40641	:255,076,055,158,160,255,128
40293	:169,004,141,136,002,032,073	40647	:152,192,219,176,024,192,130
40299	:188,246,032,225,255,240,013	40653	:193,144,006,056,233,128,197
40305	:003,076,114,254,032,163,243	40659	:076,230,158,192,065,144,052
40311	:253,032,024,229,173,021,083	40665	:012,192,096,176,006,024,211
40317	:144,141,033,208,173,022,078	40671	:105,032,076,230,158,169,225
40323	:144,141,032,208,173,020,081	40677	:000,153,000,162,136,192,104
40329	:144,141,134,002,108,002,156	40683	:255,208,217,152,192,128,107
40335	:160,169,004,162,004,160,034	40689	:176,024,192,096,144,006,111
40341	:004,032,186,255,032,192,082	40695	:056,233,032,076,013,159,048
40347	:255,162,004,032,201,255,040	40701	:192,065,144,012,192,091,181
40353	:160,255,200,192,087,240,015	40707	:176,008,024,105,128,076,008
40359	:014,185,000,002,201,045,102	40713	:013,159,169,000,153,000,247
40365	:208,244,169,171,153,000,094	40719	:160,136,192,255,208,217,159
40371	:002,208,237,169,198,141,110	40725	:169,008,141,020,162,169,178
40377	:000,003,169,157,141,001,144	40731	:020,141,008,160,096,006,202
40383	:003,032,115,000,076,156,061	40737	:169,013,032,210,255,096,040

40743 :032,146,145,032,189,255,070
 40749 :032,146,145,132,251,133,116
 40755 :252,032,146,145,169,002,029
 40761 :162,008,168,032,186,255,100
 40767 :032,192,255,032,080,145,031
 40773 :176,009,169,251,166,101,173
 40779 :164,100,032,216,255,169,243
 40785 :002,032,195,255,076,055,184
 40791 :158,169,001,133,212,165,157
 40797 :045,133,251,096,095,000,201
 40803 :000,000,000,000,000,000,099
 40809 :000,000,000,000,000,000,105
 40815 :000,000,000,000,000,000,111
 40821 :000,000,000,000,000,000,117
 40827 :000,000,000,000,000,021,144
 40833 :255,255,255,255,255,255,123
 40839 :255,255,255,255,255,255,129
 40845 :255,255,255,255,255,255,135
 40851 :255,255,255,255,255,255,141

40857 :255,255,255,255,255,255,147
 40863 :255,000,000,000,000,000,158
 40869 :000,000,000,000,000,000,165
 40875 :000,000,000,000,000,000,171
 40881 :000,000,000,000,000,000,177
 40887 :000,000,000,000,000,000,183
 40893 :000,000,000,000,255,255,186
 40899 :255,255,255,255,255,255,189
 40905 :255,255,255,255,255,255,195
 40911 :255,255,255,255,255,255,201
 40917 :255,255,255,255,255,255,207
 40923 :255,255,255,255,255,000,214
 40929 :000,000,000,000,000,000,225
 40935 :000,000,000,000,000,000,231
 40941 :000,000,000,000,000,077,058
 40947 :069,084,065,066,065,083,163
 40953 :073,067,032,080,076,085,150
 40959 :083,148,227,123,227,067,106

JACKSON

LA BIBLIOTECA CHE FA TESTO

HOME E PERSONAL COMPUTER

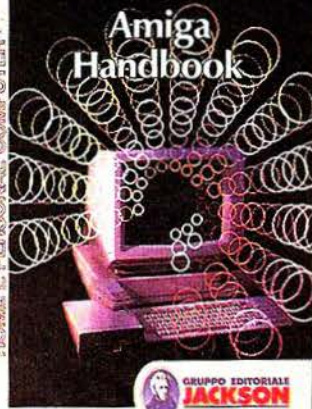
D. Lawrence, M. England
AMIGA HANDBOOK
 Pagine 189, L. 35.000
 Cod. CC320

Ancora una volta il Gruppo Editoriale Jackson si dimostra tempestivo e al passo con i tempi. A distanza di pochi mesi dalla presentazione al pubblico di Amiga, il nuovo computer della Commodore, a tutti i possibili utilizzatori di questa macchina viene rivolta una pratica e utilissima pubblicazione. Nel corso del testo sono analizzati il microprocessore 68000 e

David Lawrence - Mark England

Amiga Handbook

HOME E PERSONAL COMPUTER



il coprocessore, l'ampia serie di librerie interne che controllano la grafica, il suono, le animazioni e la capacità di Amiga di parlare.

Il libro affronta inoltre l'interfaccia Intuition (addetta al controllo delle finestre e delle icone), la Comand Line Interpreter di testo, i programmi in amigaBASIC, l'ambiente Workbench e la sua versatilità operativa, l'uso e i comandi dell'editor di schermo. Insomma, una guida preziosa per chi vuol conoscere tutto su questo versatile computer.

UN NUOVO FANTASTICO PROGRAMMA TI ASPETTA IN EDICOLA



C64 & 128 · MSX · C16 & Plus 4

**SONO
IN EDICOLA
CON
CASSETTA**

LA SOFTRIVISTA JACKSON PER GLI UTENTI C16 & Plus 4

**JACKSON
SOFT**

compilation

C16

ANNO 2 N.3 MARZO

LA SOFTRIVISTA JACKSON PER GLI UTENTI C64 & 128

**JACKSON
SOFT**

compilation

C64 & 128

ANNO 2 N.3 MARZO

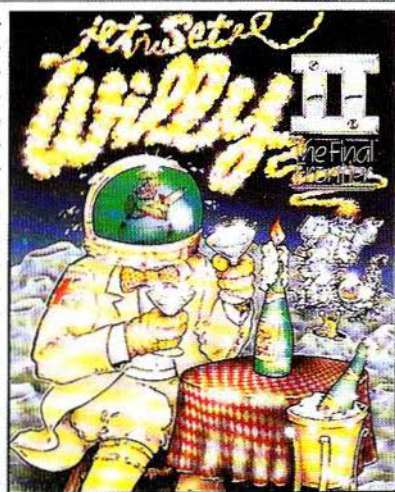
LA SOFTRIVISTA JACKSON PER GLI UTENTI MSX

**JACKSON
SOFT**

compilation

MSX

ANNO 2 N.3 MARZO/APRILE 1987 L. 8000



SUPERGAME:

JET SET WILLY II pag. 4

GUIDA ALL'INPUT pag. 9

LISTATI:

AGGUATO SUL FONDO pag. 10

DISASSEMBLER pag. 14

ARTICOLI:

CORSO PRATICO DI LINGUAGGIO MACCHINA pag. 18

PICCOLI TRUCCHI pag. 19



**GRUPPO EDITORIALE
JACKSON**